# Automatische videogebaseerde analyse van de prestatie van operatoren in mixed-model-assemblagestations

# Automated Video-Based Analysis of Human Operators in Mixed-Model Assembly Work Stations

Karel Bauters

Promotoren: prof. dr. E.-H. Aghezzaf, em. prof. dr. ir. H. Van Landeghem
Proefschrift ingediend tot het behalen van de graad van
Doctor in de ingenieurswetenschappen: bedrijfskundige systeemtechnieken
en operationeel onderzoek

UNIVERSITEIT
GENT

Vakgroep Industriële Systemen en Productontwerp
Voorzitter: prof. dr. E.-H. Aghezzaf
Faculteit Ingenieurswetenschappen en Architectuur
Academiejaar 2018 - 2019

ISBN 978-94-6355-253-0
NUR 953
Wettelijk depot: D/2019/10.500/61

# Examination board

## Chair of the Examination Board

Prof. Dr. Ir. Gert De Cooman

## Voting Members

Prof. Dr. Johannes Cottyn

Prof. Dr. Veronique Limère

Prof. Dr. Dieter Claeys

Prof. Dr. Ir. Sofie Van Hoecke

Prof. Dr. Trijntje Cornelissens

## Supervisors

Prof. Dr. El-Houssaine Aghezzaf

Prof. Em. Dr. Ir. Hendrik Van Landeghem

# Dankwoord

Het behalen van een doctoraat is een lange weg die soms bezaaid ligt met hindernissen en obstakels. Gelukkig moest ik deze weg niet alleen afleggen. Daarom wil ik mijn oprechte dank betuigen aan een aantal mensen, zonder wie dit nooit mogelijk was geweest.

In de eerste plaats wens ik mijn promotoren Hendrik Van Landeghem en El-Houssaine Aghezzaf te bedanken voor hun vertrouwen om mij deze kans te geven. Bij hen kon ik altijd terecht voor hulp en advies wanneer ik het spoor even bijster was.

Verder ook een welgemeend woord van dank aan Johannes, om me de tijd en ruimte te geven om deze opdracht tot een goed einde te brengen.

Ook de leden van de jury wens ik te bedanken voor hun uitdagende vragen, kritische opmerkingen en suggesties die de kwaliteit van dit proefschrift sterk verbeterd hebben.

Reeds van dag één heb ik mij altijd thuis gevoeld op de vakgroep in Zwijnaarde. Ik wens daarom Veerle, Peggy en Kurt te bedanken voor alle administratieve, logistieke en technische hulp die ze steeds met alle plezier geven. Ook de vele collega's en ex-collega's die ik doorheen de jaren mocht leren kennen: Derek, Benedikt, Onne, Cedric, Ihsan, Alessandro, Matthias, Wouter, Matthieu, Birger, Stijn, Rodrigo en vele andere. Bedankt voor de toffe babbels, de steunende woorden en vernieuwende inzichten. Ook wil ik de collega's in Kortrijk bedanken om mij het laatste jaar met open armen te ontvangen. De toffe sfeer zorgt ervoor dat ik telkens met plezier naar Kortrijk afzak.

Jens, Kenric, Glenn, Tom, Sofie, Dieter, Robin, Wouter, Senne, Thijs, Natan, Jorrit, Rani, Lars, Aico, alle voetbalvrienden en nog zo veel anderen: bedankt voor alle muzikale, sportieve en andere leuke momenten die een welgekomen afwisseling waren!

Een speciaal woord van dank wil ik richten aan mijn ouders. Bedankt voor alle kansen die jullie me gegeven hebben, voor alle steun en voor me altijd met raad en daad bij te staan wanneer het nodig was. Zonder

jullie was dit nooit gelukt! Ook Arvid, Janna, Femke en Kenny wil ik bedanken voor alles wat zij voor mij hebben gedaan de afgelopen jaren.

Ten slotte is er nog één iemand die ik heel uitdrukkelijk wil bedanken. Micheline, bedankt voor alles. Reeds zes jaar lang kan ik altijd bij jou terecht. Weten dat ik altijd op jou kan vertrouwen, doet enorm veel deugd. Bedankt voor de vrolijke noot die jij in mijn leven brengt, ook al is dat niet altijd even gemakkelijk.

# English Summary

## Introduction

In manufacturing industries in general and more specifically in assembly companies, there is a clear trend towards mass customization. Small batches of high varieties of products need to be produced by the production system with “batch-size-one production at the cost of mass production” as the ultimate goal. This requires highly flexible, reactive and adaptive production systems which are inherently complex. Despite the developments in the field of flexible automation, the required flexibility level of these production systems is often achieved through the use of human operators. Human operators possess cognitive capabilities which cannot be matched by the intelligence of current automated systems and therefore they are much more proficient in reasoning and making critical decisions.

## Problem statement

There is a clear need for methods that are capable of providing accurate and up-to-date information on the performance of manual assembly work stations. Current time study and methods engineering techniques are not sufficient anymore to suit the needs of today’s complex and highly dynamic production environments.

Video images have been used by methods engineers for years, but they are typically still manually analyzed. Video processing techniques have evolved over the last decade and make it possible to automatically retrieve information from video images. The use of video cameras for operator monitoring provides a number of advantages. Cameras are less intrusive for the operator than other monitoring systems based on wearable technology, cameras have become relatively inexpensive over the last years and they are a source of rich information about the process.

For these reasons, we investigate how multi-camera monitoring systems can be used to facilitate time and method study techniques

and drive continuous improvement on the shop floor level in modern flexible assembly environments.

## Approach

The proposed monitoring framework uses the output of video processing algorithms developed by researchers of IMEC. These algorithms provide accurate information about the operators' location throughout his work cycle. This video analysis framework is based on the visual hull concept to create a 3D model of the operator based on silhouettes provided by the different cameras using voxel carving. Silhouettes are extracted using foreground/background segmentation techniques. Industrial settings however, are very dynamic environments. One of the main issues when using cameras in these kind of environments, are occlusions. Many different static and moving objects might block the view on the operator partially or even completely. To overcome these issues, occlusion maps were introduced to model the occluded areas in every cameras viewpoint. These occluded areas are not taken into account when reconstructing the model of the human operator.

The resulting raw trajectories generated by the video processing algorithm still contain a significant amount of noise. We propose a data processing procedure to clean up the raw output data of the video system and give meaning to the resulting trajectories. The data processing framework makes use of an annotated work station layout to segment trajectories into useful work cycles or tasks. It enables automatic generation of time standards and work instructions which can be documented with the captured video images.

The use of video images makes it possible to monitor a work station for a prolonged period of time. Manual observations only provide snapshots of the situation and often lead to Hawthorne effects, meaning that the operator adjusts his/her behavior because of the presence of an observer. However, analyzing long streams of video images is time-consuming. Therefore we developed an off-line unsupervised clustering methodology that aims to separate normal

operator behavior from anomalous events. By pointing the analyst directly to possible problems or issues in the work station, the analysis time can be significantly decreased.

The proposed framework is based on Dynamic Time Warping (DTW) to assess the similarity of different work cycles, because of DTW's capability to handle time deformations in time series. To cluster trajectories, traditional agglomerative clustering (AHC) methods were used. A statistical permutation testing method was developed to eventually decide which trajectories should be considered as outliers or anomalies and which represent normal behavior, based on the similarity structure provided by the AHC methods.

The output of the off-line clustering algorithm is used to generate base models for normal patterns, which serve as the input for a real-time work cycle classification and outlier detection framework. The real-time methods proposed rely on the Keogh lower bound concept for DTW to match newly incoming operator trajectories to these predefined base models. This real-time classification and outlier detection can be used to provide contextualized information to the operator in the work station and allows for immediate reaction when disturbances are detected.

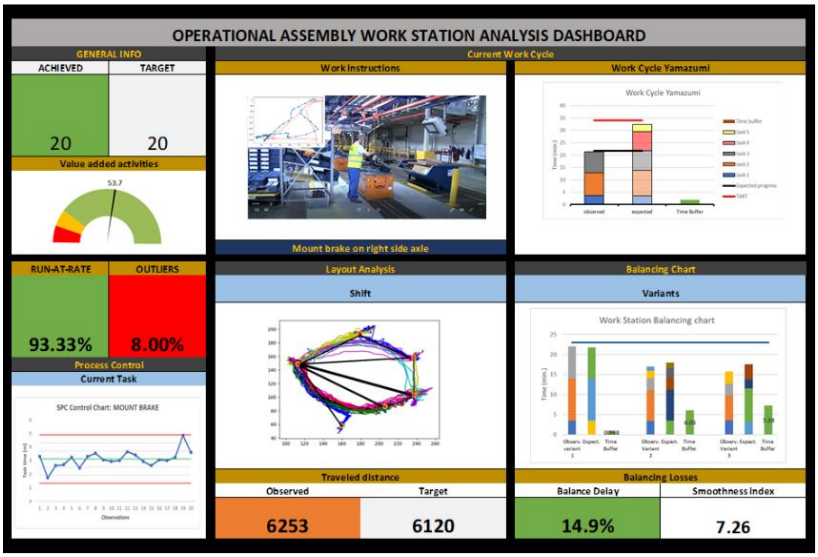
In order to provide this information to operators and production managers in a way they can use it to analyze and improve their own processes, an operational assembly work station analysis dashboard is developed. The dashboard can be customized to the needs of the user in order to facilitate root-cause analysis and drive continuous improvement on the shop-floor level. The dashboard incorporates a number of KPI's that are directly relevant on the work station level and makes use of various automatically generated charts and graphs to provide transparency in the assembly process.

An example of this dashboard is shown in the figure below.

## Conclusions

In this dissertation we presented a methodology to automate the analysis of mixed-model assembly line work stations, making use of a

multi-camera based monitoring system. Automated information extraction from operator trajectories and trajectory classification methods were developed to facilitate the monitoring of assembly line work stations and provide real-time support to drive continuous improvement actions on the shop-floor level.




The methods presented in this dissertation were tested and validated on a number of experimental and industrial use cases. The results of these experiments are promising. The off-line clustering method is capable of classifying operator trajectories with an accuracy and precision of over 90%. With this, the method clearly outperforms more traditional clustering methods. What is even better, if mistakes were made, these were typically normal trajectories which were considered to be anomalous. The risk of not detecting possible problems is thus kept to a minimum.

An experimental proof-of-concept of the real-time anomaly detection method shows that this method is capable of detecting issues with high accuracy. With a maximum calculation time per frame of 0.07 seconds, real-time problem detection appears to be possible, knowing that a framerate of 2 frames per second seems to be the minimum to

achieve acceptable classification results. The results of these methods were incorporated in an operational assembly work station analysis dashboard which supports production managers and assembly operators in their pursuit of operational excellence.

The real-time monitoring and work station analysis framework proposed, could fundamentally change the way work study and continuous improvement of assembly line work stations is approached. On the level of the assembly station and assembly line, operators and production managers are enabled to analyze and rethink their processes by providing him with relevant real-time information and a very reactive feedback loop on both the work station and assembly line level respectively.

The proposed framework also changes the role of the industrial or methods engineer. Some of the tasks typically performed by the methods engineer, such as setting time standards and generating work instructions, are (partially) taken over by the real-time monitoring system. These tasks are almost impossible to perform manually in current flexible high variety production environments. Instead, his main tasks now consist of disturbance handling and validating changes that originate from the shop floor level. This means that the engineer is now capable of managing multiple work stations/lines at once and react much faster. This reactivity is essential to stay competitive in today's market.

# Nederlandstalige samenvatting

## Inleiding

Bij maakbedrijven in het algemeen en meer specifiek de assemblagebedrijven, is er de laatste decennia een duidelijke trend naar maatwerk. Er moeten almaar kleinere hoeveelheden van een grotere variëteit aan producten geassembleerd worden. Stukproductie aan de kostprijs van massaproductie is daarom het ultieme doel van hedendaagse productiebedrijven. Om dit te bereiken zijn flexibele, reactieve en adaptieve productiesystemen noodzakelijk, wat de complexiteit van dergelijke systemen enkel maar vergroot. Ondanks de vele ontwikkelingen om machines en automatisering flexibeler te maken, wordt er nog vaak gerekend op operatoren om de nodige flexibiliteit van het productiesysteem te garanderen. De menselijke intelligentie en het vermogen om te redeneren kan nog niet geëvenaard worden door de huidige generatie van geautomatiseerde systemen.

## Probleemstelling

Het dynamische karakter van de hedendaagse markt zorgt voor een duidelijke nood aan methodes en systemen die in staat zijn om accurate en actuele informatie over het assemblageproces te genereren. De huidige analysemethodes in het domein van werk- en methodestudie voldoen niet langer aan de eisen van hedendaagse productiesystemen.

Reeds vele jaren maken methode-ingenieurs gebruik van videobeelden. De analyse van deze beelden gebeurt weliswaar meestal manueel en vergt daarom heel veel tijd. In de laatste decennia werd heel wat vooruitgang geboekt in het domein van automatische videoanalyse, wat het mogelijk maakt om heel wat informatie automatisch te extraheren uit deze beelden. Het gebruik van videocamera’s biedt een aantal voordelen. Video-opnames

kunnen vanop een afstand gemaakt zonder de operatoren te hinderen of extra te belasten, camera's zijn relatief betaalbare sensoren en de resulterende beelden bevatten meer informatie dan andere monitoring systemen die vandaag de dag gebruikt worden.

Om deze redenen onderzoeken we in dit proefschrift de mogelijkheden die multi-camera systemen bieden om de analyse van assemblagestations te vereenvoudigen en verbeteringsacties op de werkvloer te ondersteunen.

## Methode en aanpak

Het voorgestelde monitoring systeem maakt gebruik van video-analyse algoritmes die ontwikkeld werden door onderzoekers binnen IMEC. Hun beeldverwerkingsalgoritmes zijn in staat om de locatie van de operator doorheen de volledige werkcyclus op een accurate manier te berekenen. Deze algoritmes maken gebruik van het 'visuall hull' concept om een 3D model van de operator te genereren voor elk frame in de videostream. Hiervoor wordt het silhouet van de operator geëxtraheerd uit de beelden van elke camera afzonderlijk en wordt er vervolgens 'voxel carving' gebruikt om de 3D-reconstructie stap voor stap op te bouwen. Deze techniek kent echter een aantal uitdagingen die de toepassing in dynamische omgevingen zoals assemblagestations bemoeilijken. Het grootste probleem is occlusie. In industriële omgevingen kunnen zowel statische als bewegende voorwerpen het zicht van bepaalde camera's op de operator belemmeren. Dit leidt tot inaccurate silhouetten die uiteindelijk leiden tot onvolledige modellen. Om dit probleem op te lossen, wordt een methode voorgesteld die gebruik maakt van occlusiekaarten die voor elke camera de zones waarin het zicht belemmerd is, modelleren. Deze zones worden niet in rekening gebracht wanneer het uiteindelijke model wordt opgesteld.

De resulterende ruwe trajecten die gegenereerd worden door de video-analyse methodes, bevatten ruis die de uiteindelijke analyse kunnen verstoren. Daarom werd een dataverwerkingsmethode opgesteld die deze ruis verwijdert uit de ruwe trajecten en betekenis geeft aan deze trajecten. De dataverwerkingsmethode maakt gebruik

van informatie over het werkstation om de lange stroom aan videobeelden op te delen in bruikbare cycli of taken. Verder is de methode in staat om standaard tijden en werkinstructies automatisch te genereren. Hierbij kunnen de originele videobeelden gebruikt worden om deze instructies beter te documenteren.

Het gebruik van camera's zorgt ervoor dat er voor een langere tijd gemeten kan worden. Manuele analyses en observaties zijn vaak slechts een momentopname. Regelmatig treden tijdens de observaties zogenaamde Hawthorne-effecten op, waarbij operatoren hun gedrag aanpassen omdat ze geobserveerd worden. De manuele analyse van lange video opnames is echter en tijdrovende taak. Daarom hebben we een off-line clustering methode ontwikkeld die ons in staat stelt om normaal gedrag van operatoren te onderscheiden van verstoringen en problemen. Op deze manier kan de manuele analyse beperkt worden tot enkel deze stukken van de opname die noodzakelijk zijn.

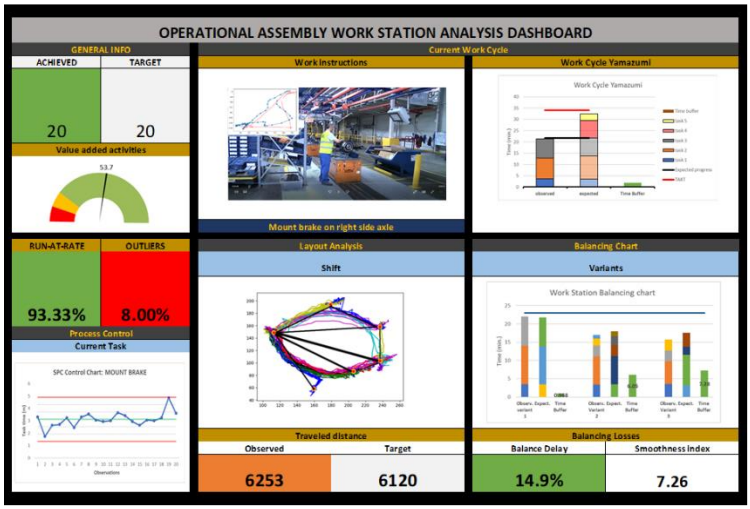
De voorgestelde methode vergelijkt verschillende trajecten door middel van "dynamic time warping" (DTW). Deze methode wordt reeds jaren gebruikt in spraakherkenning omwille van zijn capaciteit om om te gaan met tijdsvervormingen van tijdsseries. Hiërarchische clustering wordt gebruikte om een data set te ordenen op basis van de gelijkenissen tussen trajecten. Om finaal de beslissing te nemen welke trajecten er niet normaal beschouwd kunnen worden, werd een statistische permutatietest gebruikt ontwikkeld.

De resultaten van de off-line clustering methode worden gebruikt om modellen van normale patronen op te bouwen. Deze modellen worden gebruikt als input voor een real-time classificatiemethode. Deze methode stelt ons in staat om online normale trajecten en fouten te detecteren. De methode is gebaseerd op een ondergrens berekening voor DTW, zoals deze werd voorgesteld door Keogh. Deze methode kan gebruikt worden om de informatie naar de operator te voorzien van context en maakt het mogelijk om snel in te grijpen wanneer mogelijke problemen gedetecteerd worden.

Om deze informatie over te brengen naar operatoren en ploegbazen op een manier dat zij deze kunnen gebruiken om hun eigen processen

te analyseren en verbeteren, werd een operationeel dashboard voor assemblagestations ontwikkeld. Het dashboard kan aangepast worden aan de noden van de gebruiker en maakt het op die manier gemakkelijker om oorzaken van bepaalde problemen op te sporen. Het dashboard omvat een aantal prestatie-indicatoren die rechtstreeks betrekking hebben op de taken die uitgevoerd worden in het werkstation. Door middel van een aantal schema's en grafieken wordt een beter inzicht in het assemblageproces verschaft.

Een voorbeeld van het dashboard wordt weergegeven in onderstaande figuur.




## Conclusie

In deze verhandeling wordt een methode voorgesteld om de analyse van assemblage werkstations te automatiseren en verbeteren. De voorgestelde methode maakt gebruik van een systeem met meerdere camera's om operatoren te volgen doorheen hun werkcyclus. De resultaten van dit onderzoek werden samengebracht in een operationeel dashboard dat de operatoren en ploegbazen ondersteunt en continue verbeterprocessen op de werkvloer aanwakkert.

De methodes en algoritmes die voorgesteld worden in dit proefschrift, zijn getest en gevalideerd op een aantal experimentele in industriële cases. De resultaten van deze experimenten zijn veelbelovend. De off-line clustering methode is in staat om trajecten van operatoren te classificeren met een accuraatheid en precisie van meer dan 90%, wat beter is dan meer traditionele clustering methodes. De fouten die gemaakt worden door de methode bestaan typisch uit normale trajecten die door het algoritme als verdacht worden beschouwd. Het risico om echte fouten niet te detecteren, werd tot een minimum herleid.

Het real-time foutdetectie algoritme werd experimenteel getest. Uit deze testen blijkt dat fouten met een heel hoge kans gedetecteerd worden, terwijl de maximale rekentijd per frame tijdens deze testen beperkt bleef tot 0,07 seconden. Wetende dat er slechts 2 frames per seconde nodig zijn om accurate classificatie te doen, kunnen we concluderen dat deze methode zijn doel bereikt.

De real-time monitoring methodes voorgesteld in dit proefschrift, veranderen de manier waarop methodestudie en continue verbeterprocessen worden aangepakt binnen de productieomgeving. Op het niveau van het werk station leveren deze methodes actuele en accurate informatie aan die operatoren en ploegbazen in staat moet stellen om hun eigen processen te analyseren en waar nodig bij te sturen. Het dashboard zorgt ervoor dat het effect van elke aanpassing heel snel teruggekoppeld wordt.

Ook de rol van de methode-ingenieur zal door het gebruik van deze methodes gevoelig veranderen. Het monitoring systeem automatiseert een deel van de taken die typisch door ingenieurs worden uitgevoerd. Voorbeelden zijn het meten en opstellen van standaard tijden en genereren van werkinstructies. Deze taken zijn in de huidige omstandigheden te tijdrovend om nog manueel uit te voeren. Door deze uit de handen van de methode-ingenieur te nemen, verschuiven zijn taken meer naar het oplossen van problemen en valideren van nieuwe methodes die ontstaan vanop de werkvloer. Hierdoor kan de ingenieur een overzicht houden over de volledige lijn en kan hij veel sneller reageren wanneer nodig. Deze verhoogde

reactiviteit is essentieel om competitief te blijven in de huidige marktomstandigheden.

# Table of contents

# List of Tables

# List of Figures

# List of Acronyms

## A

**AHC** Agglomerative Hierarchical Clustering
**ATCGD** Adaptive Trajectory Clustering based on Grids and Density

## B

**BoL** Border of Line

## C

**CCD** Charge coupled device
**CNC** Computer numerical control
**CPPS** Cyber Physical Production Systems
**CPS** Cyber Physical Systems

## D

**DFA** Design For Assembly
**DTS** Direct Time Study
**DTW** Dynamic Time Warping

## E

**EAWS** European Assembly Work Sheet
**ERP** Enterprise resource planning

## F

**fps** frames per second

## G

**GDPR** General Data Protection Regulation

# H

**H-CPPS** Human Cyber Physical Systems
**HIM** Human Interface Mate

# I

**IE** Industrial Engineering
**IoT** Internet of Things
**IT** Information Technology

# J

**JIT** Just-in-time

# K

**KPI** Key Performance Indicator

# L

**LCSS** Longest Common Subsequence

# M

**MOST** Maynard Operation Sequence Technique
**MPC** Material planning and control
**MRP** Materials requirement planning
**MTM** Methods Time Measurement

# O

**OAWSAD** operational assembly work station analysis dashboard
**OEE** Overall Equipment Effectiveness
**OWD** One-Way Distance

# P

| | |
|---|---|
| **PCA** | Principle Component Analysis |
| **PDCA** | Plan Do Check Act |
| **PEP** | Product emergence process |
| **PMTS** | Predetermined time and motion systems |
| **PTS** | Predetermined time systems |

# R

| | |
|---|---|
| **RAR** | Run-at-Rate |
| **ROI** | Return on investment |

# S

| | |
|---|---|
| **SDS** | Standard Data Systems |
| **SPC** | Statistical Process Control |

# T

| | |
|---|---|
| **TDM** | Time Data Management |
| **TPS** | Toyota Production System |

# V

| | |
|---|---|
| **VAR** | Value Added Ratio |

# W

| | |
|---|---|
| **WS** | Work Sampling |

# Chapter 1 Introduction

In manufacturing industries in general and more specifically assembly companies, there is a clear trend towards mass customized products. Small batches of many different varieties of products need to be produced by the production system with **"batch-size-one production at the cost of mass production"** as the ultimate goal. This requires highly flexible, reactive and adaptive production systems which are inherently complex to manage.

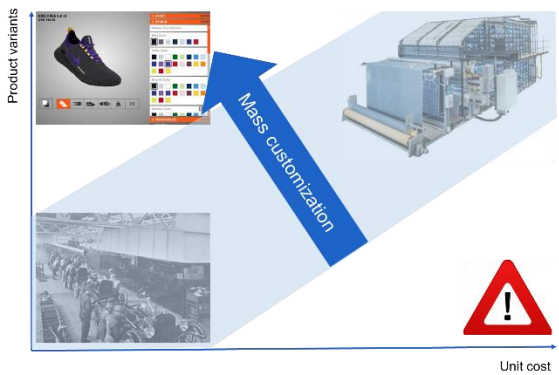



**Figure 1: Trend towards mass customization**

Despite the developments in the field of flexible automation, the needed flexibility of these production systems is often achieved through the employment of human operators. Human operators possess cognitive capabilities which cannot be matched by the intelligence of current automated systems and therefore they are much more proficient in reasoning and making critical decisions.

In order to stay competitive, flexible production companies are forced to continuously monitor, analyze and redesign their processes. One of the major challenges is the collection of reliable and detailed data about the current process. For many years, Real-time data capturing technologies, such as RFID, have already been used for some years in manufacturing environments, mainly for inventory management, planning and quality control. For manual work however, existing work measurement techniques are still relying on traditional techniques

such as stopwatch measurements and manual video analysis, making them prohibitively time-consuming. More advanced and automated techniques are therefore required to support improvement of the continuously evolving contemporary production facility.

## 1.1. Evolution of Industrial engineering methods

Over the last decades, the market for manufacturing companies has undergone a shift from mass production to mass customization. The increasing number of product variants has increased the complexity of manufacturing processes and systems. Contemporary production systems require adaptive and reactive processes. Companies need to constantly monitor, analyze, evaluate and improve their processes in order to stay competitive. Accurate and up-to-date information about the process is a strict requirement to achieve this. For years, gathering this information has been one of the main concerns of Industrial Engineers. Industrial engineering is the branch of engineering that is concerned with the design, improvement and installation of integrated systems of people, materials, information, equipment and energy. Industrial Engineering (IE) can be defined as follows [2]:

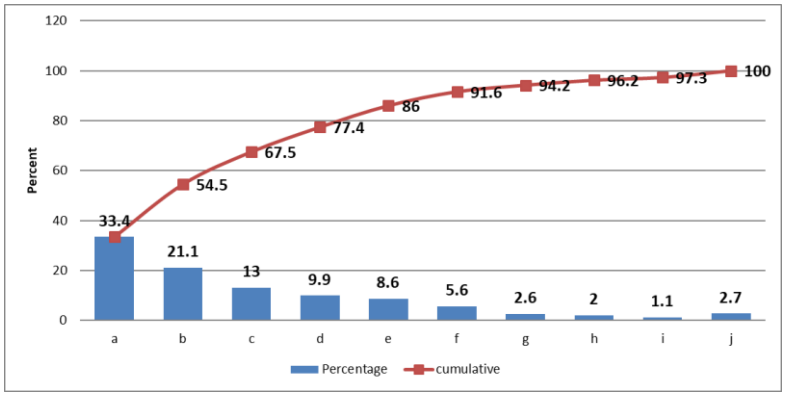
"Industrial engineering is that branch of engineering knowledge and practice which:

- **Analyzes, measures, and improves** the method of performing the tasks assigned to individuals,
- **designs and installs better systems** of integrating tasks assigned to a group,
- **specifies, predicts, and evaluates** the results obtained.

It does so by applying to materials, equipment and work specialized knowledge and skill in the mathematical and physical sciences and the principles and methods of engineering analysis and design. Since, however, work has to be carried out by people, engineering knowledge needs to be supplemented by knowledge derived from the biological and social sciences."

According to a study conducted by Miller and Schmidt [1] (Figure 2), Industrial Engineers spent more than half of their time on determining

standard times, improving methods and detecting and solving manufacturing related problems. Collecting data and making measurements accounts for a considerable amount of this time.




a) Determining a time standards for a given work task
b) Designing efficient and effective methods
c) Determining the most appropriate manufacturing operations and tooling for manufacturing a particular product
d) Analysis of manufacturing problems
e) Determining the most appropriate location for one or more facilities
f) Designing wage incentives for employees
g) Safety
h) Determining optimal reorder quantities and reorder points for inventoried items
i) Designing testing procedures to ascertain the quality level of a production process
j) Other activities

**Figure 2: Typical IE Activities [1]**

### 1.1.2. Work study

Work study is the term generally used to describe all tools, techniques and methods which are used to analyze and evaluate human work. Work study aims to increase productivity by performing a systematic examination of the methods used by human operators while carrying out activities, such as assembly operations [3]. Work study consists of two types of techniques – method study and work measurement (also

called time study) – that are used to assess human work in all its contexts.

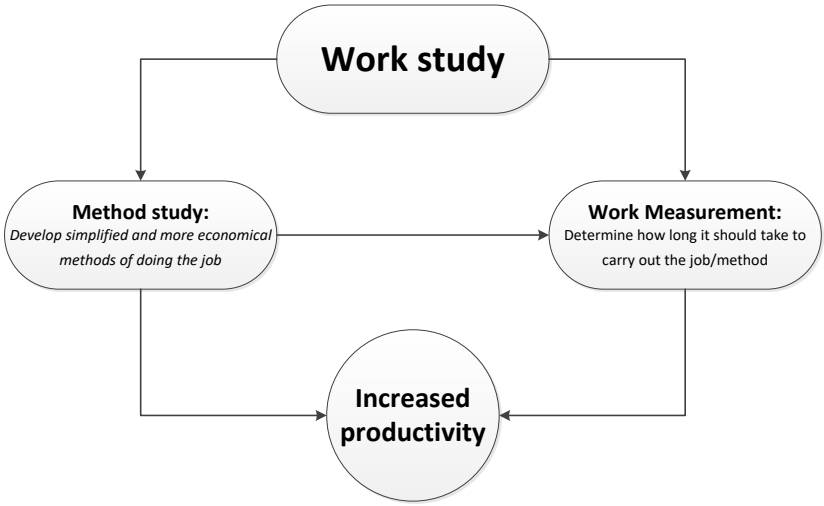



Figure 3: Work Study framework

#### 1.1.2.1. Method Study

Method study is a collection of techniques in which (human) work is critically and systematically examined and evaluated in order to establish new methods to complete the same task in a more efficient and easier way [4]. Originally, the focus of methods engineers was primarily on the analysis of human body motions involved in performing physical labor. Today, the scope is much broader. Groover [5] defines method study as the analysis and design of work methods and systems, including the tooling, equipment, technologies, workplace layout, plant layout, and environment used in these systems. Method study aims at accomplishing the following objectives [6]:

- Improving the current processes and procedures.
- Adding value to the operations under consideration or evaluation by eliminating the excess work content.
- Improving the factory and work place layout.

- Improving the use of materials, plant, equipment and manpower.
- Reducing the monotony of work to avoid mental and physical exhaustion.
- Smoothen material flow with minimal backtracking and thus improving the layout.

### *Method study techniques*

Method engineers use a systematic procedure to increase productivity [4, 7]. The steps of this procedure are described and explained below.

- **Select:** Typically, the selection of the job is done based on economic or cost effective considerations. Also quality issues, technical problems and human considerations could be reasons to select a certain job.
- **Record:** This phase involves gathering all necessary information needed for the further evaluation and examination of the job. A wide range of techniques are available for recording. The selection of the techniques used is done based on the type of work that is being studied and the level of detail that is needed.
- **Examine:** The recorded data is examined in a critical way, often through a structured questioning process. The aim of this phase is to disclose where in the process the largest improvement is situated.
- **Develop:** The aim of this stage is to identify possible improvements to the change areas defined in the previous stage. These possible solutions are then evaluated in order to develop the best or most preferred solution.
- **Install:** The success of the newly developed method is heavily depending on the way it is transferred to the shop floor. In order to return the anticipated results, the method should be thoroughly and clearly explained to those responsible for the operation. Some methods will require new skills. In that case,

the operators should be provided with the necessary training and support to acquire these abilities.

- **Maintain:** It is quite common that operators go back to older methods after a while. Therefore, it is necessary to check whether the new method is properly being followed and delivers the desired results.

Method study involves a wide variety of techniques to record, visualize and analyze operations. These techniques can be classified in three categories: charts and diagrams, motion study and work design and facility layout planning [5].

**Charts and diagrams:** The most commonly used recording techniques in method study are charts and diagrams [3]. Besides generic engineering charts and diagrams such as histograms, statistical process control (SPC) charts and Pareto charts, there are several types of IE specific charts. An overview of the most commonly used chart types in method study is given Table 1 [3].

**Table 1: Overview of IE chart types**

| | | |
|---|---|---|
| **A** | **Charts indicating process sequence** | Outline process chart |
| | | Flow process chart |
| | | Two-handed process chart |
| | | Procedure flow chart |
| **B** | **Charts using a time scale** | Multiple activity chart |
| | | Simo chart |
| **C** | **Diagrams indicating movement** | Flow diagram |
| | | String diagram |
| | | Cyclograph |
| | | Chronocyclograph |
| | | Travel Chart |

**Motion study and work design:** This field of methods engineering involves the analysis and evaluation of the basic motions a human operator uses while performing manual work. In total 17 basic motions

have been defined, most of which are related to the hands and arms [5]. The most commonly used basic motions are reach, grasp, move and release. Years of research on basic motion elements and how they affect productivity, has resulted in a set of principles on how to perform manual work in an efficient and effective manner. Barnes codified these principles of motion economy in the 1930's [8, 9]. These principles aim to eliminate wasted motion, make operator tasks easier, reduce fatigue and minimize the risk of injuries and cumulative traumas such as carpal tunnel syndrome and tendonitis [10].

**Facility layout planning:** Facility layout is an important field of study in methods engineering. The layout of a facility refers to the way functions and departments are arranged within the facility and the way materials, machines and equipment are positioned in a certain area [3, 5]. The layout has a major influence on the efficiency of the operations performed in the facility. Recently Drira et al. have conducted a literature review on facility layout planning techniques [11]. The authors classify these techniques in two categories: exact methods and approximated approaches. They perceive a clear trend in recent research papers towards the use of metaheuristics instead of exact methods.

#### 1.1.2.2. Work Measurement

Work measurement – often referred to as time study – is the technique of establishing time standards to perform certain tasks. These time standards can either be based upon the actual measurements of the work content of the prescribed method to carry out the task, estimations of the work content or historical data [7]. Additionally these time standards contain the necessary allowances to compensate for fatigue and other unavoidable delays. Meyers [12] defines time standards as the time required to produce a product at a work station fulfilling three conditions: (1) the work is performed by a qualified and well-trained operator, (2) the operator works at a normal pace and (3) the operator follows a well-specified method.

A time standard is a valuable source of information for manufacturing companies. Method engineers will use time standards to formulate solutions for typical operational problems such as:

- **Capacity:** Determining how many workers should be hired, what equipment is needed, etc…
- **Planning:** Scheduling operators and equipment so that jobs can be finished in time with the least amount of inventory possible.
- **Cost Calculation:** Calculating production costs and selling prices. Estimating production costs beforehand can be important in preparing bids for new contracts.
- **Line balancing:** Assigning the right amount of work to work stations and work cells and calculating the conveyer speed of the production line.
- **Performance measurement:** Time standards serve as the benchmark to calculate productivity and efficiency of both operators and equipment.
- **Validate methods:** Show the effect of the newly developed improved methods.

### *History of work measurement*

Although it is generally believed that Frederick Winslow Taylor is the founder of the modern method and time study, it was the French engineer Jean Rudolphe Perronet who conducted the first time studies on the manufacturing process of 18$^{th}$ century clothing pins in 1760 [7]. His work was followed by the time studies performed in 1820 by the English mathematician, philosopher and mechanical engineer Charles Whitmore Babbage, who also invented the first programmable computer. Later he also wrote the book "On the economy of machinery and manufactures", the first book on operations research in which topics such as inventory control, machine maintenance, division of labor and time study were discussed [13].

Frederick W. Taylor joined Midvale Steel Works as a worker in 1878. After working his way up through the company, he started his time study work in 1881. He was the first person to perform stopwatch measurements as we know today and, as such, he is considered the godfather of scientific management and industrial engineering. In

1911 he wrote his book “The Principles of Scientific Management”. Taylors’ scientific management consists of four simple principles [14]:

1. Develop a science for each element of a person’s work, thereby replacing the old rule-of-thumb method.
2. Select the best worker for the task and train that worker in the prescribed method developed in Principle 1.
3. Develop a spirit of cooperation between management and labor in carrying out the prescribed methods.
4. Divide work into almost equal shares between management and labor, each doing what they do best.

The effectiveness of Taylor’s method was proven in his shoveling experiment, where he experimented with different kind of shovels for different types of ore. The results of his experiment were nothing short of spectacular, as shown in Table 2 [12].

**Table 2: results of Taylors' shoveling experiments**

| | BEFORE STUDY | AFTER STUDY |
|---|---|---|
| No. People | 400-600 | 140 |
| Pounds/shovel | 3,5 - 38 | 21,5 |
| Bonus | NO | Yes |
| Work Unit | Teams | Individual |
| Cost/Ton | 7 to 8 cents | 3 to 4 cents |
| | A total savings of $78000/year | |

At the same time Taylor was developing his principles of scientific management, Henry Ford aimed to make his automobiles affordable to the masses. He incorporated the research of Taylor in the team he established to create a mass production system for the popular Model T [15]. Taylors’ concept of division of labor eventually formed the basis of Fords’ moving assembly line.

Frank and Lillian Gilbreth are known as the parents of the modern motion study technique. Motion study is a method study technique in

which the body motions used to perform a certain operation are evaluated in order to improve processes by eliminating unnecessary motion and simplifying the necessary body motions. Frank developed the idea of motion study during his time as a bricklayers' apprentice. He noticed that his instructor was using a different set of motions when he was working by himself compared to when he was explaining Frank how to lay bricks. Frank Gilbreth was convinced that there could only be one best method to lay bricks. In his search for the best method, the theory of motion study arose. He was able to increase the average number of bricks per hour from 120 to 350 [7]. Later on Frank and Lillian, a trained psychologist, went into consulting and developed numerous method study techniques such as micromotion study, process charts, the cyclograph and chronocyclographs. They also studied the effects of fatigue, monotonous work and skill transfer.

The work of Taylor was continued by two of his associates. Carl G. Barth introduced the concept of allowances in work measurement. Henry L. Gantt developed a performance control system based on simple charts that would measure performance while visually comparing it to the projected schedules. Gantt charts are still used in industrial engineering and project management today.

Later on in the 20$^{th}$ century, Dr. Ralph M. Barnes became one of the first and probably best-known professors in the field of Industrial Engineering. He published a thorough description of Gilbreths' micro-motion study and developed the procedure for work sampling. He also performed a great number of method studies based on video images and created a number of videos for training time study engineers in the field of pace rating [12]. Video images as an analysis tool were introduced Marvin E. Mundell with the development of memomotion, a stop-action filming technique used to determine time standards [16].

Shigeo Shingo was a Japanese consultant who, among other things, worked for Toyota. Inspired by Taylor's Principles of Scientific Management, he is considered to be an important contributor to some concepts of the Toyota Production system (TPS), which was created by Taiichi Ohno. Shingo was the first to document the TPS in his book "A study of the Toyota Production System from an engineering

viewpoint". This way, he lay the basis for lean manufacturing as we still know it today.

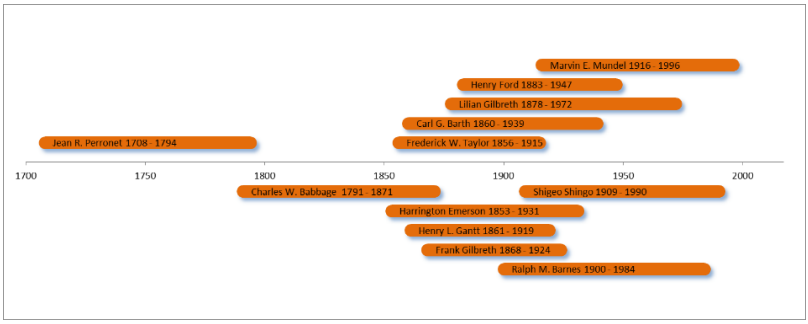



**Figure 4: Pioneers of Work Study: timeline**

## *Time standards in contemporary manual assembly environments*

According to Wacker and Sheu [17], manufacturing planning and control are two of the most important aspects to consider when improving the performance of manufacturing systems. With the development of the first MRP (material requirements planning) system in the 1960's, the first computers and digital information systems were introduced in the planning and control of production environments [18]. In today's era of Digital Manufacturing, the use of digital tools and models in production planning, control and optimization seems to be a matter of course. The effectiveness of these digital information systems is heavily depending on the data and information quality the system is provided with [19]. In that context, a number of studies have identified the availability of valid data as the critical factor for the successful implementation and operation of enterprise resource planning systems (ERP) [20, 21] .

During the 1980's and 1990's, the use and significance of time data in manufacturing companies was consistently reduced [22]. However, more recently we have experienced a revival of time data management. In the last decade, manufacturing companies, research

institutes and non-profit organizations all recognized the importance of time data as a key factor in their decision making process [23].

Time data can be used throughout the complete product emergence process (PEP), starting from the product design phase until the production phase [24]. When referring to assembly processes specifically, time data is used for assembly-oriented product design. The Design for Assembly (DFA) approach, as proposed by Boothroyd [25], uses time data to estimate the production cost of different design alternatives from the early product design phase on. In the process planning phase, time data is mainly used for work station and production system design. As an example in assembly environments, we can mention the European assembly work sheet (EAWS) which uses, amongst other data, manual assembly time data to assess ergonomic risks of an assembly process [26]. On the operational level, time data is mainly used for monitoring and short-term optimization purposes.

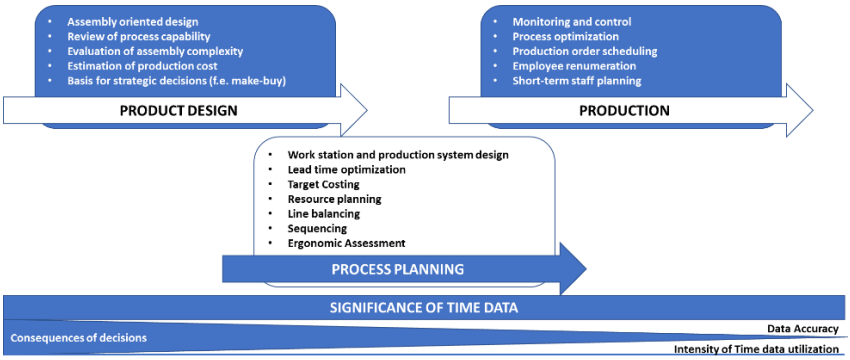



**Figure 5: use of time data throughout PEP (adapted from[24])**

Figure 5 indicates how time data is used in the different phases of the PEP. Time data can be determined iteratively, starting from time estimations in the early product design phase. As the PEP progresses, the accuracy and use of time data will increase. However, decisions made based on time data in the early phases, will have a significantly larger impact on the final production cost.

Despite the clear importance of having up-to-date time data of production processes, only limited attention has been paid in literature to the gathering and management of accurate time data of manual assembly processes, as most researchers assume that this data is readily available. However, in reality, companies are often hesitant to implement real-life time data in the MPC systems. In a study in over 60 manufacturing companies, Almström and Kinnander [27] found out that most of the time data in the MPC systems was incorrect and only 25% of these companies ever updated their process times after they had been put into the MPC system. Kuhlang [28] identifies the fact that gathering time data is time-consuming and expensive as the main reason for the gap between real production times and the data used in MPC systems.

Despite the trend towards more automated production processes, manual assembly remains the most cost-effective method in the context of high variety low-volume production. According to El Maraghy [29], assembly tasks account for around 50% of the total production time and 20% of the total production cost. In automotive industry, 20 to 70 percent of the total production time is comprised of manual assembly tasks. Production times, especially manual task times, are not static. Manual assembly times evolve as the product and production process design changes over time. Also, manual assembly times are depending on the capability of the operator, his/her experience level and even the level of fatigue.

Acquiring accurate timing data and keeping this data up-to-date is crucial and requires a rapid and precise measurement tool. However, most companies are often stuck with traditional stopwatch and clipboard methods for work measurement. When these tools are used in today's high variety, low-volume production environments, they fail miserably. As mentioned earlier, the cost of these methods is one reason for their failure. However, there are more reasons why traditional stopwatch measurement methods are not suitable for current manufacturing environments. Best [30] identified 8 barriers, technological or social in nature, that prevent these traditional methods from quickly acquiring accurate time data (Figure 6).

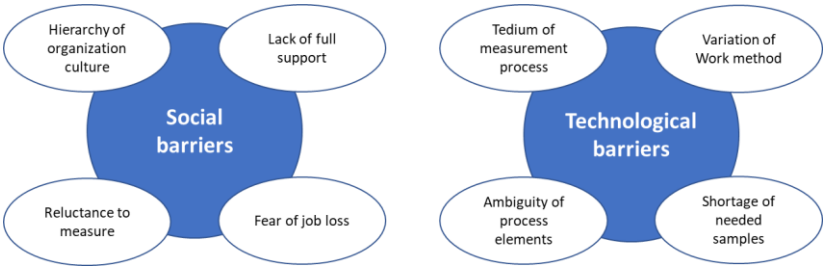



**Figure 6: Work measurement barriers in modern manufacturing companies**

**Social Barriers**

**Hierarchy of Organization Culture:** Many companies still use a rigid hierarchical management approach. New initiatives are often imposed on employees in a top-down manner. When work measurement is performed by Industrial Engineers without involving employees, experienced operators might feel that their knowledge and expertise is ignored. In this case, operators will have difficulties to support the imposed time standards. This approach also undermines any form of support for new work measurement initiatives.

**Lack of full support:** As mentioned above, an effective work measurement system needs to be supported by everyone in the organization. However, in many cases management perceives the acquisition of accurate time data as a cost factor with little or no direct return on investment (ROI). Furthermore, management often fears the resistance of employees when setting up a work measurement system. Because of the difficulty to quantify benefits and the fear that the system will create more harm than good, management often discards work measurement programs.

**Reluctance to measure:** Because work measurement techniques fell into disuse in the 1980's, time standards were solely based on educated guesses. Operators rate their own performance against these inaccurate estimates. In many cases, this leads to an overestimation of the performance. Since they have been using the same methods for years and the company managed to survive, employees don't see the need to change the evaluation process and therefore they are often reluctant to the implementation of more fact-based systems [30].

**Fear of job loss:** The main goal of gathering time data is identifying which processes or tasks need to be improved. Often, operators experience this as a direct threat to their job security, as they believe that the measurements will be used to evaluate them and test whether they can keep up with certain expectations or target speed. For this reason, operators will sometimes try to hinder the measurements or change their behavior to influence the results. This effect of changing behavior when under observation is often referred to as the Hawthorne effect.

**Technological Barriers**

**Tedium of the Measurement Process:** One of the main problems with traditional work measurement techniques is the effort it takes from the Industrial Engineer to perform the observation and the analysis of the results. Especially in a high-variety low volume production environment where certain tasks are only performed rarely. An observer completing a traditional time study would have to be available throughout the whole work day to capture these events. Also the analysis is a time-consuming process. In many cases the observer needs to transfer his/her written observations to a digital format to perform statistical analysis of the results.

**Variation of work methods:** In high-variety production environments, different operators tend to deviate from the standard work method. According to Nadler [31] such variations are disastrous for the prediction value of a time study using traditional methods.

**Ambiguity of process elements:** Unless the observer is very familiar with the task under investigation, he or she will find it difficult to determine which elements of the task are independent of each other. Independence in this context means that the variability of one work element is unrelated to the variability of other elements. In practice that means that there should be clear and defined break points that define the start and end of a specific work element. The definition of these separate work elements is important. If complete work cycles would be timed as a whole and the results show a lot of variation in task time, this variation is probably caused by a highly variable work

element. If the definition of these elements is not done properly, there is no way to explain the variability of the time estimates.

**Shortage of samples:** Traditional work measurement relies on statistical analysis to provide accurate and significant time estimations. The more samples the observer collects, the more accurate the time estimation will be. In high-variety production environments where the number of different tasks and the variability on the task times are huge, the number of samples needed to obtain reliable time estimations is too large to acquire manually.

### *Work measurement techniques*

As explained in the previous section, accurate and up-to-date production time data and time standards are still very important in today's production environments. There exist a multitude of methods to determine time standards, each of them with their own characteristics and applications. According to Heinz and Olbrich [28] time determination methods can be classified in two main groups, as shown in Figure 7. The first set of methods is based on real production observations. These methods provide information on actual task times, but require extra processing to account for inefficiencies and disturbances when used to determine time standards. The second set of methods do not require measurements on the shop floor, but only provide theoretical time estimations or targets. In the remainder of this section, we will briefly present these methods, discuss their strengths and weaknesses and indicate which of the aforementioned barriers they address.

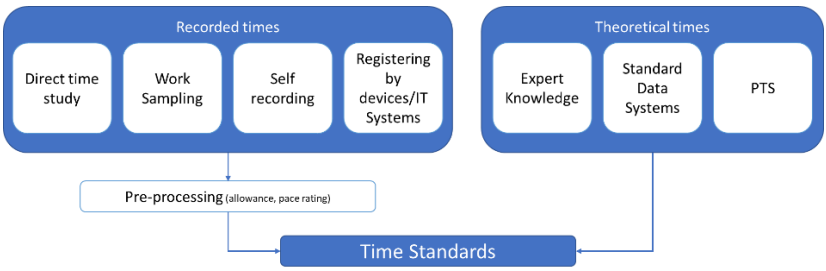



**Figure 7: classification of time standard setting techniques (adapted from[22])**

Each of these methods has its own strengths and weaknesses and operates at a specific level of the task hierarchy as shown in Figure 8 [5]. Expert opinion estimates are very rough techniques to determine time standards. The fairly low accuracy of this information increases the risk of errors in the decision making, which is not desirable [32]. Work measurement techniques are based on facts and are a more reliable way to establish time standards.

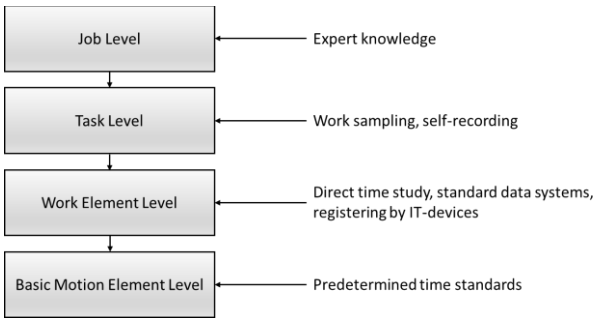



**Figure 8: Standard setting techniques and their corresponding task hierarchy level (adapted from [5])**

### Theoretical times

***Expert knowledge:*** This method is based on the experience and knowledge of people that are familiar with the concerning jobs. This person is asked to estimate the time that should be allowed for a specific job [5, 12]. This estimation is completely based on the judgment and knowledge of the expert, which makes this technique one of the least accurate and reliable [5]. The lack of observations on the other hand, removes some of the social barriers for traditional work measurement techniques.

***Standard data systems (SDS):*** Standard data systems (SDS) involve the use of elemental times obtained from earlier time studies which are indexed and stored for late use [7]. Many different parameters can cause variation in the cycle time. Standard data systems aim to discover the relation between parameter changes and their effect on the operation time. There are a number of different techniques and tools that are used in standard data systems: graphs, tables, worksheets and formulas. Multiple regression analysis is a tool

typically used for developing mathematical models that describe the influence of the different process parameters on the cycle time. Standard data systems allow one to predict the operation time before the actual production begins [33]. SDS are often used in industry for estimating labor costs for bids. Standard data enables work analysts to develop accurate time standards fast without the need for time-consuming techniques such as direct time study or predetermined time standards [12]. However, SDS are relying on historical data, therefore the accuracy of the models depends on the quality of the input data. These models are typically based on manufacturing data in manufacturing IT systems or operator self-measurement through so-called time cards. This data is often unreliable and does not take into account inefficient work methods, disturbances and operator performance.

***Predetermined time standards (PTS):*** Performance rating has always been a thorny and controversial step in direct time study. Predetermined time standards don't rely on the judgment of the analyst [5]. PTS divide manual work into basic motion elements which are each associated to a predefined normal time. Just like in direct time study, the normal times obtained through PTS require the addition of allowances to obtain fair and usable time standards. Since 1945, more than 50 different PTS have been developed and used in industry. Well-known examples of PTS are MTM-1 (Methods Time Measurement) [34], Work Factor [6] and MOST (Maynard Operation Sequence Technique) [35]. PTS databases are the result of extensive research and generate very reliable time standards. The scientific foundation of these methods improves the credibility and support for these methods by both management as well as employees [30]. Because tasks are subdivided into basic motions, the analyst does not have to be very familiar with the job. The evaluation however, usually relies on an elaborate and time consuming frame-by-frame analysis of video footage of the task. On average, it takes around 200 minutes to analyze 1 minute of video using MTM-1 [36]. The use of more condensed tables, such as MTM-3 or MTM-SAM, helps to reduce this effort (30 minutes per minute of video), but accuracy of these faster methods is significantly lower. Because of the high amount of different tasks in

high-variety production environments, one can argue whether the use of PTS is economically viable.

### Recorded times

***Direct time study (DTS):*** Traditional work measurement or direct time study (DTS) is based on a direct observation of the task. The analyst will divide the task into usable work elements and use a stopwatch or other timing device to record the time for each element. Each work element should consist of a logical group of motion elements and have a clear start and end point. A typical work element consists of all motion elements that are related to the handling of one part or object (e.g. take a part, move it to its destination and place/fix the part) [5]. During the observation, the analyst evaluates the performance of the operator in order to establish fair time standards for both the employer and the employee. This performance rating is used to adjust the results of the stopwatch study to the normal time an average worker requires to perform that task [7]. Since one cannot expect an operator to be fully productive over the course of his whole work shift, extra allowances should be added to this normal time. These allowances account for some personal time an operator needs, speed losses due to fatigue and other delays that are not caused by the operator and on which he doesn't have any influence [12]. Figure 9 visualizes how time standards are established in a direct time study.

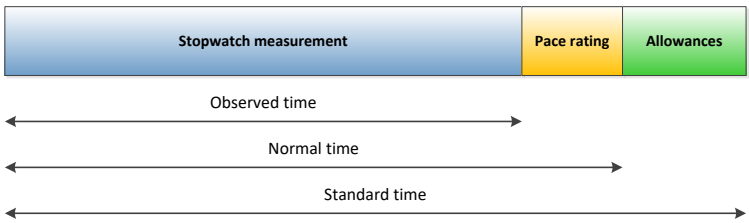



**Figure 9: from stopwatch measurement to standard time**

Just as in any other human activity, there is an inherent variability in manual work. Because of this reason work element times vary from cycle to cycle [5]. Direct time study involves a sampling procedure to overcome the issue of statistical variation in the times of these work

elements. A high sample size is expected to return very accurate results, but increases the cost of the study as well. Based on a preliminary study, the ideal number of cycles in the sample can be determined as follows [9]:

$$n = [(z)(s)/(E)]^2 \quad (1)$$

Where

**z** depends on the desired confidence interval. These values can be found in tables.
**s** is the sample standard deviation of the preliminary study.
**E** is the desired absolute accuracy.
**n** is the required number of samples.

In Table 3 we calculate the required observation time for three different work stations. The first work station only performs one single task with limited variability in the task time. The average task duration, number of different tasks and variability is increased for work station 2 and 3. Work station 3 could be exemplar for the typical work content for a work station in a large industrial equipment assembly line (atlas copco, CNHi) for one machine, not even taking into account the many different product variants. As shown in the table, the observation time required to obtain reliable data increases incrementally, making manual observations economically impossible. Note that the example only takes into account the observation time itself.

To partially overcome the cost-issue of direct time measurements, stopwatch measurements are often replaced with modern technology that is more conducive to quick and accurate data collection. Software applications such as UmtPlus® [37] and TimerPro® [38] use mobile devices to easily collect the data and automatically perform the statistical analysis using custom software based on Microsoft Excel®. However, these tools have difficulties handling unforeseen circumstances and still require an observer that is present at the work station throughout the complete observation.

Table 3: Table 3: Calculation of sample size for DTS

| | WS1 | WS2 | WS3 |
|---|---|---|---|
| **average time** | 1 | 3 | 10 |
| **stdev** | 0.1 | 0.6 | 3 |
| **Accuracy** | 0.95 | 0.95 | 0.95 |
| **different tasks** | 1 | 5 | 10 |
| **sqrt(n)** | 3.92 | 7.84 | 11.76 |
| **n** | 15 | 61 | 138 |
| **total observation time (minutes)** | 15.4 | 922.0 | 13829.8 |
| **Total observation time (days)** | 0.03 | 1.9 | 28.8 |


Figure 10: Timer Pro DTS application

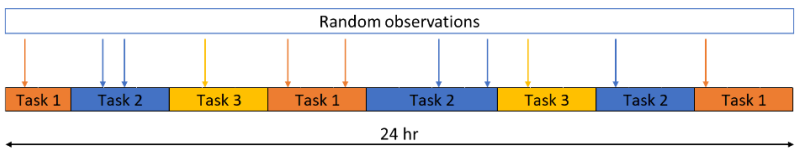
***Work sampling (WS)***: This technique consists of random observations of the operators' activities to determine how they distribute their available time over the different activities of interest [16]. Work sampling is a statistical technique in which a large number of observations are made over a prolonged period of time, and statistical conclusions are drawn about the proportion of time spent in each activity category based on the proportion of observations in that category [5]. Just like in DTS, we can reduce the statistical errors in work sampling by increasing the number of observations. The required number of observation is again depending on the desired confidence

level and accuracy [5]. Since a work sampling study results in an overview of the operators' time distribution, it is not really suited to determine time standards. However, in situations where more accurate techniques, such as PTS and direct time study, require excessive analysis times, WS can be used to calculate average task times. To do so, the total time associated with one category is divided by the total count of parts produced over that time period. This calculation is summarized in the following equation [5]:

$$T_{ci} = \frac{p_i * (TT)}{Q_i} \tag{2}$$

Where:

**$T_{ci}$** = average task time
**$p_i$** = proportion of observations associated with the category
**TT** = total time: total duration of work sampling study
**$Q_i$** = total quantity of work units associated with the work category that are completed during the total time




| Work sampling | | Direct time study |
|---|---|---|
| 2.9 hr | Task 1 | 2.7 hr |
| 3.6 hr | Task 2 | 3.3 hr |
| 2.2 hr | Task 3 | 3.0 hr |

**Figure 11: Comparison between time standards obtained through WS and DTS**

Figure 11 shows a comparison between time standards obtained through work sampling and more accurate DTS time standards. These time standards are significantly different, mainly because of the small sample size. If the number of sampling observations would increase, the outcome of the sampling study would converge to the real value. In the context of high variety production, it is often impossible to acquire a high number of samples of a specific task, because some tasks only occur vary rarely. This makes work sampling less suitable for high-variety low-volume production environments.

***Self-Recording:*** All of the previously mentioned methods still require a significant effort to obtain accurate data. The cost of these measurements often forms a barrier for production companies to implement work measurement. To overcome this barrier, data collection can be conducted by the operators themselves [30]. Through the use of so-called tagging sheets, barcode scanners or mobile devices, employees determine separate work elements and perform the work study themselves. The operators' knowledge and expertise overcomes the issue of task ambiguity and the employee empowerment increases the support and credibility of the observations. On the other hand, self-recording can be perceived as added work load for the operator and gives the operator the possibility to influence the measurements.

***Registering through devices:*** In the last decades, many manufacturing companies started implementing novel technologies for shop-floor monitoring and control in their production systems. The data logs of these IT systems usually contain time stamps which can be used to obtain time data.

One example of such a technology is the Arkite HIM® system. The Human Interface Mate (HIM) is a standalone device equipped with a smart 3D sensor to support operators in manufacturing industry. By monitoring the assembly process, the HIM is able to detect mistakes and intervene when necessary. The HIM is also capable of collaborating with projection systems to show instructions and feedback in augmented reality on the final product. The process validation techniques used by the ARKITE HIM system, also enable capturing of production data, such as manual assembly times. However, the system requires that all production steps are a priori known and programmed into the system. The system has difficulties to handle disturbances and variable assembly sequences. Since it only uses one single sensor, the HIM is also not suited for monitoring very large assembly work stations, because the sensors' view on large parts of the work station will often be occluded by different objects in the work station.

**Figure 12: ARKITE HIM in combination with projection technology**

### *Video-based tools for method study and work measurement*

Video recordings have a long history in the manufacturing environment. It was Frank Gilbreth who was the first to introduce cameras in scientific management [39]. Gilbreth was able to overcome the doubt that followed Taylors' time studies through the use of film. Video images made these measurements replicable, more precise and open for public evaluation. In a time were movies became more popular and the working class' leisure time was increasingly spent at the cinema, Gilbreth convinced the workers to participate in his work studies by giving them the opportunity to star in their own movie. Based on the films, time measurements were made and decisions regarding new best methods were open for discussion. This way, Gilbreth's work was a big step forward in the acceptance of Taylor's ideas by operators and the scientific community [40].

Until the invention of the VCR recorders however, making videos was too cumbersome and expensive and required too much skill to be used on a large industrial scale. In addition, most attempts to use motion recordings for the purpose of productivity improvement failed because manufacturing workers feared that those recordings could be used against them sooner or later [41]. The first project for which video images were systematically used was changeover time reduction.

The advent of video tapes enabled analysts to forward, rewind and slow down video recordings which made data capturing a lot easier. The emergence of electronic spreadsheets omitted the need for manual spreadsheets and the corresponding counter intuitive time measures such as decimal minutes [41]. The cost of recording and analyzing videos kept on decreasing due to the emergence of digital

cameras, flash drives and media players. In digital videos, every frame is linked a timestamp. Today, video annotation software uses this characteristic to facilitate time and method studies. A wide variety of these software packages have been developed. They enable the analyst to break down videos into segments, label them, categorize them and analyze them. The software then uses the timestamps to determine cycle times of the total process and separate tasks. Video annotation software has a number of advantages over the traditional approach [41]:

- Video annotation automates the collection and storage of timestamps. Manually entering timestamps in spreadsheets is tedious and error-prone. Automatic retrieval of timestamps speeds up the analysis while at the same time increasing the accuracy of the time study.
- Video segments stay linked to their labels, text or other data you attach to it. This way the software can easily bring up the video segment which matches to the task or process you want to analyze.
- Annotation software can use parallel tracks to simultaneously show what several different machines and operators do at the same time.
- Most video annotation software packages offer the option to export the results into standard excels files, but they also offer easy-to-use built-in analytics.

Video annotation software is often considered to be a time study tool. The use of video data, however, should not be restricted to determining time standards. A thorough analysis of videos can help in evaluating processes, identifying and quantifying improvement potential, and designing and validating new methods. Video images contain a lot of information and are a good way to document and improve work methods [42, 43]. An example of how video images can be used as a process improvement tool can be found with Nexteer automotive. Nexteer is a global leader in advances steering and driveline systems. In 2008, they collaborated with Dartfish to introduce video analysis in their continuous improvement program. Dartfish is a world leader in 2D video analysis technology for biomechanics and is often used as a training tool for athletes, such as figure skaters. In a first test, Nexteer used the Dartfish software to record three different workers at the same operation, and then overlaid their images to show

differences in how they performed standard work such as parts set-up and quality checks. After the analysis, they achieved a 21 per cent improvement [44]. In the following years, Nexteer rolled out the video analysis software across more than 400 different processes. According to Nexteer, a typical workshop using the Dartfish technology results in a gain of 20% to 30% in operational availability, where the gain using a more traditional approach typically ranges between 5% and 7%.

### *Work measurement techniques: summary*

In this section we described the existing work measurement techniques. Each of these techniques has its specific strengths and weaknesses.

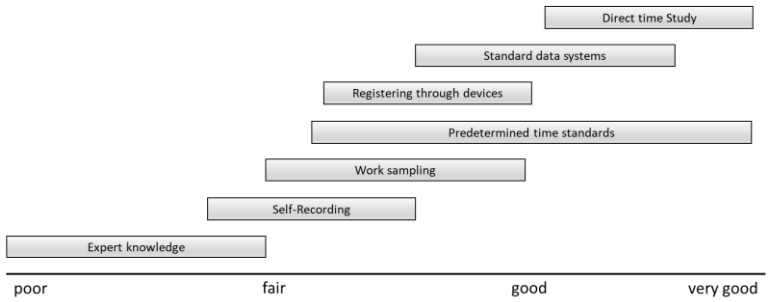



**Figure 13: relative accuracy of work measurement systems (adapted from [5])**

In [5], Groover describes the relative accuracy (Figure 13) and application speed (Figure 14) of the methods explained in this section. Work sampling is omitted in Figure 14 because it requires a prolonged period of gathering random observations. It is clear that the more accurate methods require the most analysis time. Predetermined time systems cover a wide range in both accuracy and application speed due to the plethora of different techniques used in industry. When using PTS, one always needs to make the trade-off between application speed and desired accuracy.

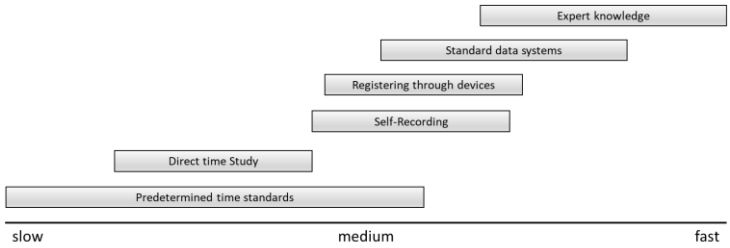



**Figure 14: Relative application speed of work measurement systems (adapted from[5])**

Despite their relatively high accuracy, PTS do not provide a good basis for performance measurement and monitoring. Disturbances, inefficiencies and variability can only be discovered through the analysis of observed data. DTS is still the most accurate work measurement methods. Yet to be economically viable in todays' manufacturing environments, more automated observation and analysis methods based on advanced technologies are required.

Video cameras and image recognition could facilitate the analysis of manual assembly and manufacturing tasks. Video images are being used in method study and work measurement for years. However, most video-based methods and software tools still rely on manual analysis of the images. By partially automating the analysis of video footage, valuable and accurate production data could become available with less effort.

## 1.2. Towards Industry 4.0

Throughout history, manufacturing industry has undergone a number of major revolutions and paradigm shifts. Most of these shifts were based on novel and game changing technologies. The first industrial revolution began at the end of the 18$^{th}$ century and was characterized by mechanical production plants based on steam engines. These steam engines made the transitions from hand production to machines possible, increasing the productivity of manufacturing plants enormously. A well-known example of these developments is the weaving loom used in textile industry. At the beginning of the 20$^{th}$ century, the advent of electrical energy, internal combustion engines and communication technologies initiated the second industrial

revolution, characterized by mass production lines. The third revolution was a digital transformation. Around 1970's manufacturing companies started automating their production processes with the help of novel electronics and information technologies [45].

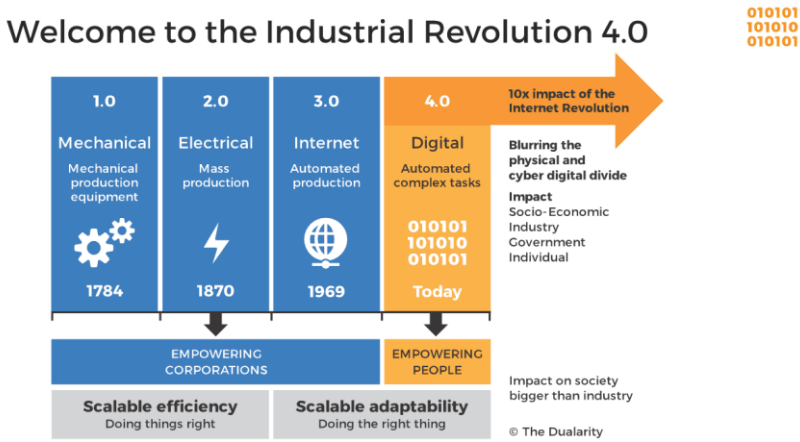



**Figure 15: evolution of manufacturing industry [46]**

Currently, manufacturing companies are undergoing a fourth revolution, commonly referred to as Industry 4.0. The initial concept for Industry 4.0 was proposed by the German government in 2011 as a development program for the German economy [47]. Industry 4.0 transfers the principles of the Internet of Things on to the production environment. At the basis of Industry 4.0 are the so-called cyber physical systems (CPS). They are systems of collaborating computational entities which are in intensive connection with the surrounding physical world and its on-going processes, providing and using, at the same time, data-accessing and data-processing services available on the Internet [48]. Cyber physical production systems (CPPS) combine the latest and foreseeable advances in computer science and information and communication technologies with manufacturing science and technology to fulfill the agile and dynamic production requirements and to increase the effectiveness and efficiency of the manufacturing industry.

So far, CPPS research and development is mainly focusing on technological advances that enable Industry 4.0, such as the

development of cheaper sensors. The involvement of the operator however is greatly underappreciated. Consequently, there is a risk that novel technologies outpace the adaptive capacity of their users, in this case manufacturing operators. This would result in a gap between the state-of-the art manufacturing system and operators that are unable to perform well in collaboration with the system. To have a disruptive impact on organizations, it is necessary that operators trust the CPS and AI-techniques [49]. The importance of self-explaining systems is demonstrated through the growing interest of industry and academics in developing explanatory AI technologies and solutions [50]. Furthermore, the importance of operator involvement becomes obvious in a study performed in Belgium in 2016. This study states that an average of 7% of the operators is absent on any given day. Moreover, 3% of the operators working in manufacturing companies are absent for more than one year. On the other hand, the study also showed that motivated and involved operators were 50% less likely to be absent and are 16% more productive [51]. These results are confirmed in a poll performed by the German Innovation Center for Industry 4.0, where manufacturing managers were asked about the main challenges they encounter in implementing Industry 4.0 in their production environment. Over 35% of the respondents identified employee involvement as their main challenge.

The results of these surveys show a clear need for the development intelligent operator workspaces. Human-Cyber-Physical systems (H-CPS) make use smart sensor systems and wearable devices to integrate the human operator into flexible and multi-purpose manufacturing systems [52].

### 1.2.1. The human operator in Industry 4.0 manufacturing

As the manufacturing systems have evolved over time, also the tasks of the operators have undergone major changes. Until the first halve of the 20$^{th}$ century, Operator 1.0 was mainly performing manual tasks, supported by some limited mechanical tools. With the advent of numerical controlled machines (f.e. CNC) and enterprise information systems, the tasks of the operator changed into more assisting tasks. The operator 3.0 generation works in close collaboration with machines and even robots (cobots). In future human-cyber physical

systems, operator 4.0 will be a smart and skilled operator that performs tasks, assisted by machines and information systems when needed. The aim is to develop adaptive automation that seamlessly interacts with operators and enables them to make full use of their unique capabilities [53].

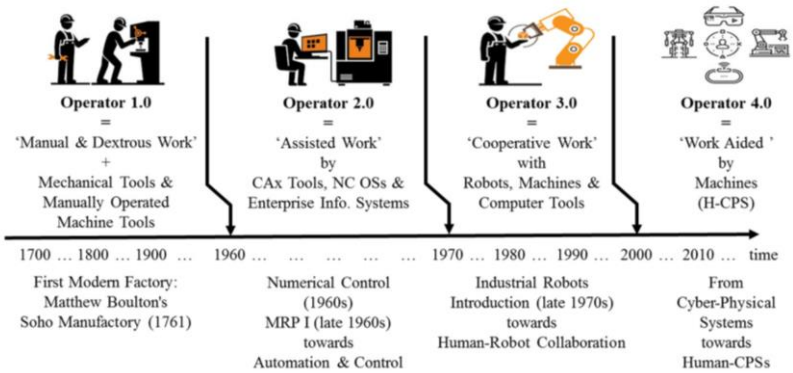



**Figure 16: Evolution of operator tasks in manufacturing environments [53]**

The operator 4.0 framework for future work place design entails many different aspects important for human operators. Analytical operator solutions based on big data analytics help to collect, organize and analyze large datasets created by numerous sensors and monitoring systems. This information supports the operator in his decision making process [54]. This information can be transferred to the operator through Augmented Reality techniques to make information transfer easier and less time-consuming. Healthy operator solutions make use of information coming from hearth-rate sensors, cognitive load sensors and activity trackers to safeguard the operators well-being. Smarter operators seamlessly interact with all other resources in the production system as well as with the information systems to provide information to support their work and decision making process. Social operators use methods developed in social robotics and intuitive human-machine interfaces to interact with smart factory resources. Interaction with virtual models of the work station provides advantages when it comes to training operators and helps to reduce change-over times between different tasks. Most of the aspects mentioned above have an impact in the mental of cognitive capabilities of the operator. Super-strength operator systems, such as exoskeletons, aim to enhance to physical capabilities to allow the

operator to perform an extended range of manual tasks. The collaborative operator on the other hand is able to work in close interaction with cobots which take over tasks that are ergonomically or cognitively challenging for the human.


**Figure 17: Operator 4.0 concept in Industry 4.0 manufacturing environment**

### 1.2.2. IoT-Based solutions for operator activity tracking

Smart sensors form the basis of CPPSs, as they are able to provide real-time information on the state of the production resources [55]. The operator 4.0 concept aims to create human-cyber-physical production systems (H-CPPS) that enhance the operators' capabilities thanks to the dynamic interaction between human operators and production systems based on context-aware information flows. In these H-CPPS, smart sensors the key enabling components that allow accurate context gathering through real-time operator activity tracking [56].
Usually, Radio Frequency Identification (RFID) technology is used to monitor operator activity [57]. RFID sensors can be used to track motion of operators and machine activity as well as material movement [58] and the flow of workpieces [59, 60] through the production system. These RFID sensor systems enable on-line measurement of task times, waiting times and production lead times. They even unveil useful information about resource and operator utilization and can help to measure critical performance indicators, such as overall equipment effectiveness (OEE). Based on this

information, shop floor control (SFC) and process optimizations can be realized.

As wearable sensors are becoming more and more accessible, they are being used more frequently in real-world applications. Various examples of the application of wearable sensor systems for human operator activity tracking in manufacturing environments have been presented in literature. Koskimaki et al. [61] described a case study in which they used a wrist-worn inertia sensor to recognize four types of basic activities: hammering, screwing, spanner use and the use a power drill for screwing. In their study they were able to detect these activities with an accuracy of over 90%. The monitoring of these activities enables the development of proactive systems: f.e. the operator can be given specific instructions when performing a specific tasks. Furthermore, this monitoring system can be used as a quality control system by checking whether all necessary actions are performed before sending the product to the next work station. A similar application was proposed by Ward et al. [62], where 2 body-mounted accelerometers were combined with microphones to develop a sensor system that recognizes typical activities in a wood shop. Stiefmeier et al. [63] aim to use wearable sensor systems to reduce the cognitive load of automotive assembly line workers. Their sensor system consists of an inertial measurement unit (IMU) mounted on the back of the palm of the operators' hand, a number of force-sensitive resistors (FSR) placed on both arms and an RFID reader mounted between the thumb and the index finger. This last sensor recognizes when a specific RFID-tagged tool is utilized. Their sensor system was integrated in a motion suit which the operators had to wear while performing assembly tasks.

Most of the sensor systems proposed, require the operator to wear specific sensors or suits which may cause hindrance when performing their tasks. Vision-based tracking systems are less intrusive. On top of that, video images contain more and richer information than any other smart sensor used in aforementioned applications. Despite the clear potential, the use of computer vision in manufacturing remains very challenging. Production environments are rather dynamic environments. Typical stereo-vision applications [64, 65] are not able to cope with objects occluding the view of the sensor on the object that is being tracked. Networked smart-camera systems can help to overcome this issue, but the large amount of data generated by these systems can make the transmission and storage of this data difficult

[66]. The constant development of novel IoT and decentralized computing technologies however, strengthen our believe that multi-camera operator monitoring systems could be a valuable source of contextualized assembly process information.

### 1.2.3. IoT based solution for operator support

It is clear that smart IoT sensors facilitate the collection of critical data from the operator. This information however, only becomes really valuable when it is used to provide real-time operator support. When operator activities are recognized in real-time, contextualized work instructions, warnings and information on his performance can be shown to the operator. This enables the operator to make better decision, resulting in more efficient, safer and more qualitative processes.

Real-time information can be shown to the operator using fixed screens, mobile devices (tablets, smartphones) or wearable devices such as smart glasses. The design of intuitive operational dashboards, apart from the device used, increases the performance of the operator by speeding up the information transfer [67]. The advent of wearable devices increased the flexibility and efficiency of real-time information transfer by combining the hands-free operation of fixed screens with the mobility of mobile devices. As an example, Airbus claims the introduction of smart AR glasses on their final assembly line helped them to reduce the installation time of the flight test equipment on the A330Neo aircraft from one complete day with 3 operators to 6 hours for one operator [68]. Similar results were reported in [69], where smart glasses were used in the assembly of heavy agricultural equipment at AGCO. By presenting context-aware instructions for next assembly steps and information about quality inspection procedures, learning rate of new operators was reduced with 50%, inspection time by 30% and 25% decrease of the overall production time was achieved. These results are supported by an extensive evaluation of 385 existing use cases in German industry [70]. This study proves that the operator 4.0 framework works in practice and has many advantages: (1) elimination of paper-based administration; (2) real-time feedback about the manufacturing process from and to the operator; (3) increased productivity; (4) decreased number of errors; and (5) operator training time reduction.

### 1.2.4. Towards I4.0: summary

It is clear that the implementation IoT technologies will result in more efficient and smarter workplaces. Production systems will become safer, more controllable and manageable than ever before. Especially in complex high variety low volume manufacturing settings, these benefits will be clearly visible. Despite the development in the field of flexible automation, the human operator will still play a significant role in future production systems, as also stated by Elon Musk when talking about the excessive automation in the Tesla manufacturing process.


**Figure 18: Elon Musk: "excessive automation was a mistake"**

In a study in over 215 use cases [51] where human operators are involved, better employee support and information transfer are considered to be the most important benefits of Industry 4.0 concept. By providing contextualized real-time information, IoT technology will empower human operators to make well-funded decisions and enable them to exploit their unique capabilities to their full potential.

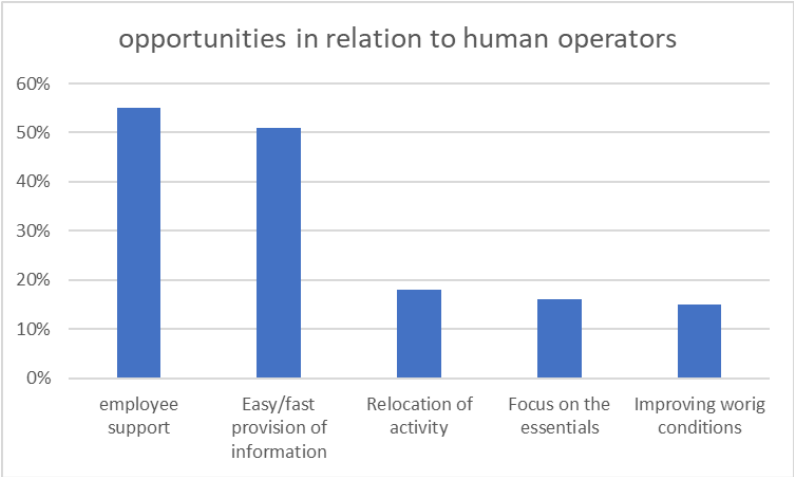



**Figure 19: Industry 4.0 benefits for human operators [51]**

According to this same study, one of the main challenges to exploit the opportunities provided by IoT technologies to their full potential lies in the monitoring and recognition of human operator activities.

## 1.3. Discussion and research gap

In the last decades, the market demand for manufacturing companies has undergone a shift from mass production to mass customization. Increasingly more variants of products are being produced in ever smaller batch sizes. This shift increases the complexity of manufacturing systems, often resulting in less efficient and productive processes. To counteract the performance losses, manufacturing systems need to be flexible, adaptive and reactive. This can only be achieved when accurate and up-to-date information about the manufacturing processes is available.

Despite the trend towards more automated processes, human operators still play a significant role in these flexible production systems. Human workers possess unmatched cognitive capacities which makes them the most flexible asset in modern production systems. The advent of IoT technologies creates opportunities to provide accurate and real-time information about the production processes. Mobile or wearable devices can be used to feed this information back to the operator. This way the operator receives the necessary contextualized information that empowers him to make

well-funded decisions. This results in smarter operators that receive the necessary support to use their strengths to their full potential.

There is a clear need for methods that are capable of providing accurate and up-to-date information on the performance of manual assembly work stations. Current time study and methods engineering techniques are not sufficient anymore to suit the needs of today's complex and highly dynamic production environments.

Video images have been used by methods engineers for years, but they are typically still manually analyzed. Video processing techniques have evolved over the last decade and make it possible to automatically retrieve information from video images. The use of video cameras for operator monitoring provides a number of advantages. Cameras are less intrusive for the operator than other monitoring systems based on wearable technology, cameras have become relatively inexpensive over the last years and they are a source of rich information about the process.

For these reasons, throughout this PhD thesis, we investigate how multi-camera monitoring systems can be used to facilitate time and method study techniques and drive continuous improvement on the shop floor level in modern flexible assembly environments.

## 1.4. Research Questions

This doctoral dissertation investigated the use of multi-camera based operator monitoring systems to automate the analysis process and support assembly operators and production managers in their pursuit of operational excellence and continuous improvement. The main research questions that are tackled in this dissertation are the following:

- **How can normal video cameras be used to provide useful data on the behavior and performance of human operators in mixed-model assembly work stations?**

- **How can machine learning techniques be used to speed up and improve the work station analysis process, typically performed by industrial engineers.**

- **What are the key performance indicators in manual assembly processes and how can they be extracted from the video images.**

- **To what extent is it possible to provide all stakeholders in the process with real-time support for continuous improvement and operational excellence on the shop floor level?**

## 1.5. RESEARCH OUTLINE AND CONTRIBUTIONS

This section provides an overview of the structure of the remainder of this dissertation and the main research contributions:

In CHAPTER 2, we present a multi-camera based monitoring framework for assembly operators. The framework makes use of video processing techniques developed by IMEC researchers, to derive trajectories followed by the operator during his work cycle. In this chapter we propose a **data processing procedure** to clean up the raw output data of the video system and give meaning to the resulting trajectories. The data processing framework makes use of an annotated work station layout to segment trajectories into useful work cycles or tasks. It enables (partial) automatic generation of work instructions which can be documented with the captured video images. Furthermore, the framework is capable of generating up-to-date time standards. This way, the proposed framework relieves the methods engineer of some typical time-consuming tasks.

In CHAPTER 3 we describe an **operator trajectory clustering method**. The aim of this method is to distinguish normal operator behavior from anomalous events. Trajectories that represent similar tasks are automatically grouped together. Outliers or anomalous events are automatically detected and presented to the methods engineer for further investigation. This way, the proposed framework is capable of monitoring work stations for a prolonged period of time and automatically highlights the sequences which are relevant for more detailed investigation to speed up work station analysis process.

An empirical study, based on a number of different experimental data sets, was performed to determine the most suitable combination of

clustering method and similarity measure for this application. The proposed algorithm uses dynamic time warping to calculate the similarity between trajectories because of its' capability to deal with time deformations. The classification itself is based on normal hierarchical clustering methods. Hierarchical clustering however only provides insights in the similarity structure of a data set, therefore probabilistic permutation testing is used to automate the final clustering step. The proposed method reaches an average precision and accuracy that exceeds 90% based on the experimental data sets used in this research. This exceeds the performance of more commonly used classification methods. For every resulting cluster, average sequences are also calculated which serve as base models for the real-time trajectory classification method presented in the following chapter.

CHAPTER 4 aims to describe how the results of previous chapters can be used in real-time to drive continuous improvement on the shop floor level and create a highly reactive feedback loop. Based on the results of the clustering method presented in chapter 3, a **real-time work cycle classification and outlier detection** procedure is proposed. The method uses the Keogh lower bound concept for DTW to compare live trajectories to the previously calculated base models. This way, incoming sequences that are not matched to one of these models are classified as an outlier which is send to the methods engineer for further investigation. The proposed procedure is capable to process a new video frame every 0.07 seconds, which is more than sufficient knowing that only 2 frames per second are actually needed to provide accurate classification results.
The second section of this chapter focusses on real-time decision support on the shop floor level. In this section an **operational assembly work station analysis dashboard (OAWSAD)** is presented. Dashboards are being used more and more in manufacturing environments, however there is a clear lack of dashboards that drive and accelerate continuous improvement on the work station or shop floor level. The OAWSAD presented in this section aims to bridge this gap by integrating real-time performance data and automatically generated charts and diagrams that have proven their value in industrial engineering and continuous improvement for many years. The

proposed dashboard is customizable to the needs of the user. This way it facilitates root-cause analysis and drives continuous improvement processes on the shop floor level. In that respect, it distinguishes itself from existing manufacturing dashboards typically used today.

CHAPTER 5 presents the conclusions of this research. The answers to the research questions described above, are summarized. Also the limitations of this research and perspectives for further research are discussed.

## 1.6. PUBLICATIONS

The results of this research are published in international journals and the proceedings of a number of international peer-reviewed conferences. The complete list of all publication is provided below.

**Journal publications**

- Bauters, Karel, Cottyn, J., Claeys, D., Slembrouck, M., Veelaert, P., & Van Landeghem, H. (2018). Automated work cycle classification and performance measurement for manual work stations. (A. Sharon, Ed.)*ROBOTICS AND COMPUTER-INTEGRATED MANUFACTURING*, *51*. (**A1**)

- Bauters, Karel, Govaert, T., Limère, V., & Van Landeghem, H. (2015). Forklift free factory: a simulation model to evaluate different transportation systems in the automotive industry. *INTERNATIONAL JOURNAL OF COMPUTER AIDED ENGINEERING AND TECHNOLOGY*, *7*(2), 238–259. (**A2**)

**Peer reviewed conferences – first author**

- Bauters, Karel, Cottyn, J., & Van Landeghem, H. (2018). Real time trajectory matching and outlier detection for assembly operator trajectories. In P. Gerril (Ed.), *Proceedings of 16th the International Industrial Simulation Conference*. Presented

at the International simulation conference, Eurosis. **(Best paper award)**

- Bauters, Karel, Van Landeghem, H., Slembrouck, M., Van Cauwelaert, D., & Van Haerenborgh, D. (2014). An automated work cycle classification and disturbance detection tool for assembly line work stations. *International Conference on Informatics in Control, Automation and Robotics, Proceedings* (Vol. 2). Presented at the International Conference on Informatics in Control, Automation and Robotics (ICINCO - 2014).

- Bauters, Karel, Van Landeghem, H., Van Haerenborgh, D., Slembrouck, M., Van Cauwelaert, D., Veelaert, P., & Philips, W. (2013). Multi-camera complexity assessment system for assembly line work stations. *The European Concurrent Engineering Conference, Proceedings*. Presented at the The European Concurrent Engineering Conference (ECEC - 2013). **(Best paper award)**

- Bauters, Karel, De Cock, K., Hollevoet, J., Dobbelaere, G., & Van Landeghem, H. (2016). A simulation model to compare Autonomous vehicle based warehouses with traditional AS/RS systems. *ESM 2016*. Presented at the European Simulation conference, Eurosis.

- Bauters, Karel, Govaert, T., Limère, V., & Van Landeghem, H. (2011). From bulk feeding to full kitting : a practical case in the automotive industry. *ISC'2011: 9TH INTERNATIONAL INDUSTRIAL SIMULATION CONFERENCE* (pp. 244–249). Presented at the 9th International Industrial Simulation Conference , Ghent, Belgium: Eurosis.

**Peer reviewed conferences – Co-author**

- Xie, X., Van Cauwelaert, D., Slembrouck, M., Bauters, K., Cottyn, J., Van Haerenborgh, D., Aghajan, H., et al. (2015). Abnormal work cycle detection based on dissimilarity

measurement of trajectories. *9th ACM international conference on Distributed Smart Cameras, Proceedings*. Presented at the 9th International Conference on Distributed Smart Cameras.

- Xie, X., De Vylder, J., Van Cauwelaert, D., Veelaert, P., Philips, W., Aghajan, H., Slembrouck, M., et al. (2014). Average track estimation of moving objects using RANSAC and DTW. *Eighth ACM/IEEE International Conference on Distributed Smart Cameras, Proceedings*. Presented at the Eighth ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC - 2014).

- Van Landeghem, H., Bauters, K., & Limère, V. (2013). A management system for sustainable lean implementation. *Industrial and Systems Engineering Research Conference, Proceedings* (pp. 941–950). Presented at the Industrial and Systems Engineering Research Conference (ISERC - 2013).

- Van der Wee, M., Casier, K., Bauters, K., Verbrugge, S., Colle, D., & Pickavet, M. (2012). A modular and hierarchically structured techno-economic model for FTTH deployments. *16th international conference on Optical Networking Design and Modeling, Proceedings* (pp. 1–6). Presented at the 16th International Conference on Optical Networking Design and Modeling (ONDM - 2012), New York, NY, USA: IEEE.

- Govaert, T., Bauters, K., Limère, V., & Van Landeghem, H. (2011). Forklift free factory : a case study of different transportation systems in the automotive industry. *ISC'2011: 9TH INTERNATIONAL INDUSTRIAL SIMULATION CONFERENCE* (pp. 217–224). Presented at the 9th International Industrial Simulation Conference , Ghent, Belgium: Eurosis.

- Van De Ginste, L., Goos, J., Schamp, M., Claeys, A., Hoedt, S., Bauters, K., Biondi, A., Aghezzaf, E-H., & Cottyn, J. (2019) Defining Flexibility of Assembly Work Stations Through the

Underlying Dimensions and Impacting Drivers. ICPR: 25th International Conference on Production Research, Chicago, USA

**Book section**

- Van Landeghem, H., Govaert, T., Van Landeghem, T., Van den broecke, F., Bauters, K., & Busschaert, P. (2011). *De fabriek van de toekomst ... nu! Duurzame productiviteitsverbetering voor KMO's*. Gent: Universiteit Gent - Vakgroep Industrieel Beheer

# Chapter 2 Multi-camera operator monitoring system

The recent developments in mobile computing and location-acquisition technologies have made it easier to generate trajectories of moving objects such as vehicles, animals or people [71]. Typical examples are GPS (global positioning system), Radio Frequency Identification (RFID) or Wi-Fi and Bluetooth based indoor location systems.

A *spatial trajectory* can be defined as a trace generated by a moving object. It is generally described by a series of chronologically ordered data points: $L(a_1, a_2, \ldots, a_n)$, where L is a trajectory of length n consisting of chronologically ordered points $a_1, a_2, \ldots, a_n$. Points $a_1, a_2, \ldots, a_n$ are typically described by a geospatial coordinate and a timestamp: $a = (x, y, t)$. The vision system used in this research is capable of returning three-dimensional space coordinates. However, in further experiments, the third dimension is less relevant and therefore not taken into account.

Production management and continuous improvement of production processes requires precise performance measurements and complete process visibility. In logistics, the ever increasing use of RFID technology has proven its value. Automatic and continuous object tracking can provide valuable information for production planning and control and problem detection [72] [73] [74] [59, 75].

In this chapter we present a multi-camera based operator monitoring system that is capable of generating the trajectories followed by the operator throughout his/her work cycle. The choice for video technology is based on a number of reasons: (1) Video cameras cause minimal hindrance for the operator during his work, (2) the raw video images are a rich source of information which has proven its value in industrial engineering for decades and (3) because of decreasing prices, video capturing and data storage technology has become more accessible.

This chapter is outlined as follows. In the first section we describe the experimental set-up used in this research and provide an analysis of the data sets used for developing and validating the operator monitoring and performance evaluation framework. In the second section we describe the video processing methods used to derive operator trajectories from multiple camera images. This method was developed by researchers of IMEC, but is briefly explained in this chapter to provide a better understanding of the complete framework. The output of the video processing algorithms consists of raw trajectory data (time stamped XYZ-coordinates) which contain a significant amount of noise. In the third section we present a data processing framework to filter noise and give meaning to the raw trajectory data.

## 2.1. Design of Experiments

Throughout this research, a number of different data sets have been used. This section describes how these data sets are collected. The first data set is collected through a series of experiments in a simulated work station in a laboratory setting and consist of 11 different scenarios. The second data set is a larger dataset, containing over 200 trajectories. This data set was created by monitoring people entering and leaving a research lab. The last data set was captured in a real assembly work station at a supplier of subassemblies for the automotive industry.

### 2.1.1. Experimental data set

***The assembly work station***

For the first data set, an assembly work station was simulated in our laboratory in Kortrijk. The assembly line was mimicked by a conveyer belt which brings the assembly to the work station. This conveyer belt can either by manually controlled by the operator (OD) or externally driven (ED) to assure a constant TAKT time. Parts are picked from a 2.5m wide picking rack which is equipped with a pick-to-light system and which is located 1.5m from the point where the assembly takes place. A grid was taped on the floor to validate whether the calculated

trajectories are aligned with the trajectories observed in the video images. The work station is equipped with four ceiling-mounted HD cameras, one in every corner of the scene. The avoid the risk of occlusion, one camera with a fisheye lens was mounted centrally above the workstation. An overview of the simulated work station is provided in Figure 20 and Figure 21.

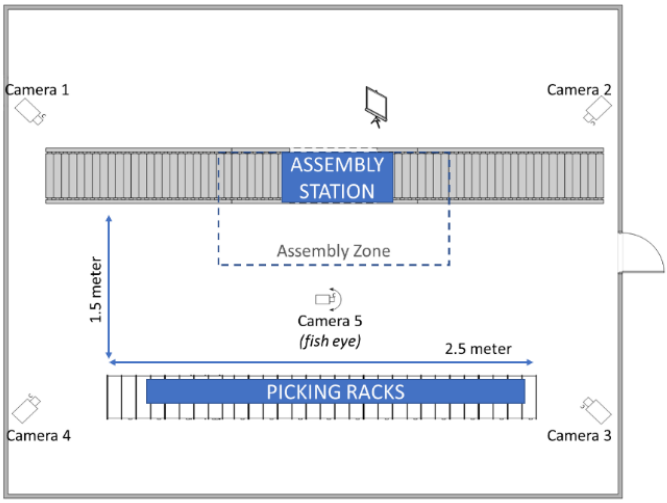



**Figure 20: Ground plan of simulated work station**

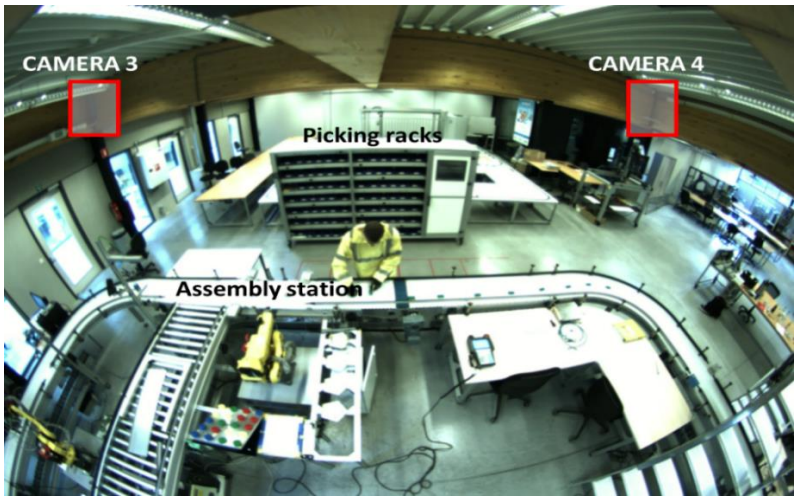



**Figure 21: Overview of the simulated work station**

***Hardware***

The cameras used for these experiments are Allied Vision Manta CCD Cameras. They have a resolution of 9 megapixel and a maximum frame rate of 125 fps. However the video processing techniques used to generate the trajectories, are not really depending on the exact type of IP cameras used. Most cameras offer a framerate and resolution which largely exceeds the requirements for the proposed methods (cfr. Chapter 3).


**Figure 22: Allied Vision Manta camera used in the experimental setup**

The video cameras are connected to a local server station to store all video data.

The data processing framework described in this chapter and the clustering and classification algorithms (chapter 3 & 4) are implemented in Python. All experiments are run on a standard Windows 10 workstation with an Intel Core I7 2.7GhZ 64 bit processor and 16GB RAM.

***Scenarios***

The parts produced in the experimental work station consist of a Duplo® base block on which a pattern of different Lego® blocks needs to be assembled. The use of Duplo and Lego blocks allows us to easily create new product variants and vary the complexity of the assembled products. An example of a finished product is provided in Figure 23.

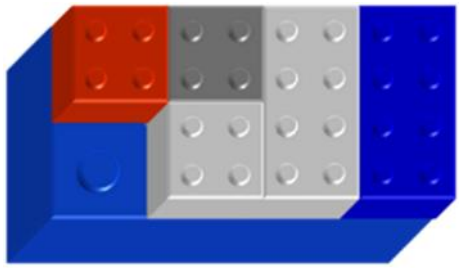


Figure 23: Example of a finished product

Based on these simple products, a number of scenarios was developed in order to validate the system and test its performance. These scenarios contain different events which are modeled after situations which can be found in real-life work stations. The different scenarios in this data set are summarized in Table 4. The parameters and events described in this table are explained below:

- **Ext_Takt:** The conveyor which delivers the base blocks to the work station can be controlled in two ways: operator driven (OD) or externally controlled (EC). OD means that the conveyor is activated by a button pushed by the worker whenever a part is finished. When externally controlled, the conveyor will deliver parts with a fixed time interval (TAKT). In that case, the TAKT is indicated in the table.
- **Variants:** The number of variants that are produced in the sequence. Custom means that every single part is unique.
- **Parts_pres:** Indicates the way parts are presented to the operator. In the one-variant scenarios parts have fixed locations in the picking rack (F). In other scenarios parts are randomly placed in the rack. Their location can be indicated by colored labels (L), small labels on the boxes themselves (I) or a pick-to-light (PtL) system is used to guide the operator.
- **Worker:** Indicates which worker performed the scenario. In total three different workers were assessed during the experiments.
- **Work cycles:** The number of different end products produced in this scenario. These scenarios only contain a limited number of work cycles because of the data storage and transfer limitations of the video system. However, since the

setting for all these scenarios is the same, different scenarios can a posteriori be merged to form larger data sets.

**Table 4: Overview of different scenarios in the experimental data set**

| Scenario | Ext_takt | Variants | Parts_pres | Worker | Work cycles |
|---|---|---|---|---|---|
| 1 | OD | 1 | F | A | 12 |
| 2 | OD | 1 | F | B | 14 |
| 3 | OD | Custom | PtL | B | 14 |
| 4 | OD | Custom | PtL | A | 14 |
| 5 | OD | Custom | PtL | C | 11 |
| 6 | OD | Custom | I | B | 14 |
| 7 | OD | Custom | I | A | 14 |
| 8 | OD | 3 | L | B | 14 |
| 9 | OD | Custom | L | B | 14 |
| 10 | OD | 2 | L | B | 14 |
| 11 | EC(25s) | Custom | L | B | 14 |

### 2.1.2. OmniLab Data Set

The second data set is a larger set of trajectories which was developed by Morris and Trivedi [76]. The dataset consists of 206 trajectories in total and contains 7 different patterns. The data set was created by recording all people walking through a research lab in one day. The video images were made using one single omni-directional camera. The resulting trajectories are depicted in Figure 24.

Although this data set does not contain trajectories of operators in industrial settings, it describes the path followed by humans in indoor environments while performing a number of tasks. Also, because of the fact that this data set is created by using vision technology to measure the behavior of the human, we believe that it can provide valuable input for the development of our classification and performance measurement framework.

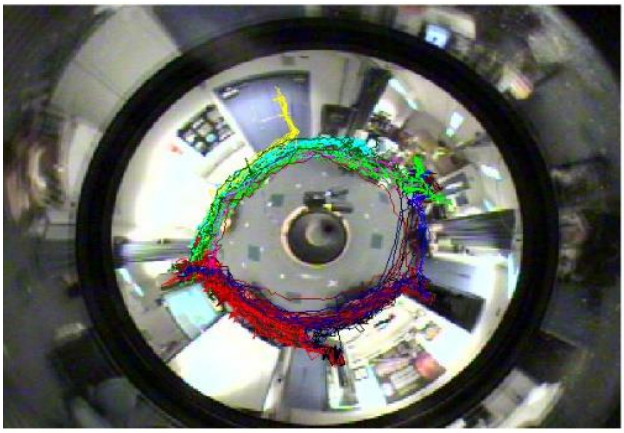


Figure 24: Overview OmniLab dataset

### 2.1.3. Industrial Case Study

This data set is generated by monitoring an assembly operator when performing assembly tasks in a work station at a supplier of subassemblies for the automotive industry. The assembly line under investigation produces rear axles for cars. In this work station, parts of the suspension system are assembled on both sides of the axle. All parts are stored in a picking rack located at the BoL. The torque wrenches used to fix the parts, are placed above the work piece. A picture of this work station is provided in Figure 25. This case study was performed as part of the IOF StarTT Project Complexity [77].

This experiment was mainly performed to test the performance and accuracy of the vision system in an industrial setting. However, the results of this experiment provide useful input to validate the classification and performance measurement framework developed in following chapters.

During this experiment, the operator was monitored using a mobile camera system consisting of only four cameras. One camera was placed on each corner of the work station. To reduce the risk of occlusions, the cameras were mounted on tripods and elevated approximately 2.5 meters above the ground. The worker was also

asked to wear a fluorescent vest for better detection by the vision system.


**Figure 25: Overview of the assembly line work station**

During the experiment, only one product variant of the axle was produced. This means that the operator had to perform exactly the same tasks in every work cycle. The operator was monitored continuously for almost 10 minutes. In total, 12 different work cycles were recorded.

Figure 26 shows the recorded trajectories this data set.

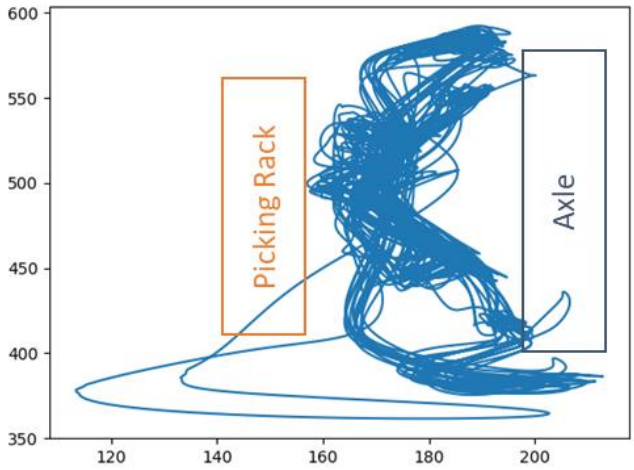



**Figure 26: Resulting trajectories of industrial use case**

In this section we describe the algorithms used to obtain the operator trajectories from the video images. These algorithms are developed by researchers of IMEC. However, to create a better understanding of the complete monitoring system, we briefly explain them below.

### 2.2.1. Visual hull principle

To obtain the trajectory of operators during their work cycle, a 3D representation of the operators body is generated for every frame of the video sequence. Afterwards, the center of mass of this 3D representation is calculated and taken as the operators' position. Every second, the cameras each generate 20 images, which means that we can calculate the operators' position every 50 milliseconds. The trajectory thus consists of a set of xyz-coordinates of the operators' location which are chronologically ordered.

The 3D representation of the operators body is generated using the visual hull concept proposed by Laurentini [78]. Since their introduction, Shape-from-silhouettes algorithms as these techniques are also called, have been used by many researchers to obtain 3D models from multiple 2D images [79, 80]. The basic idea behind these algorithms is the following: first the 3D-space under consideration is divided in a 3-dimensional grid pattern. The cubical cells of these grids are called voxels and can best be compared to pixels in 2D-image representations. Then, the silhouette of the followed object is extracted from the 2D-view of every camera and a 3D infinity cone with the camera as the apex and the extracted silhouette as the base, is generated. This infinity cone is basically a collection of the voxels in which the object might be situated, given the information one single camera can provide.

**Visual hull algorithm:**

---

**Input:** camera calibration,foreground/background masks
**Output:** voxelated visual hull

```
for all voxel in voxel_space do
        while voxel occupied and not all camera views evaluated
do
                lookup voxel's projection on next camera view
                if projection is foreground then
                        classify voxel as occupied
                else
                        classify voxel as unoccupied
                end if
        end while
end for
```

---

**Figure 27: Visual hull algorithm**

For the extraction of the operators silhouette from the video images, foreground-background segmentation is used. The idea behind this approach is to detect moving objects by evaluating the difference between the current frame in the video stream and a reference frame, often called "background image" or "background model".

The 3D model of the operator is then generated by carving away voxels that are not within the infinity cones of all cameras. It is therefore easy to understand that the quality of the 3D model increases with the number of cameras added to the system. The visual hull concept is explained in Figure 28. In this figure the blue circle is the object that is being tracked. The resulting visual hull consists of the object and all parts indicated in gray. It is clear that the accuracy of the visual hull method increases with number of cameras that are added to the system.

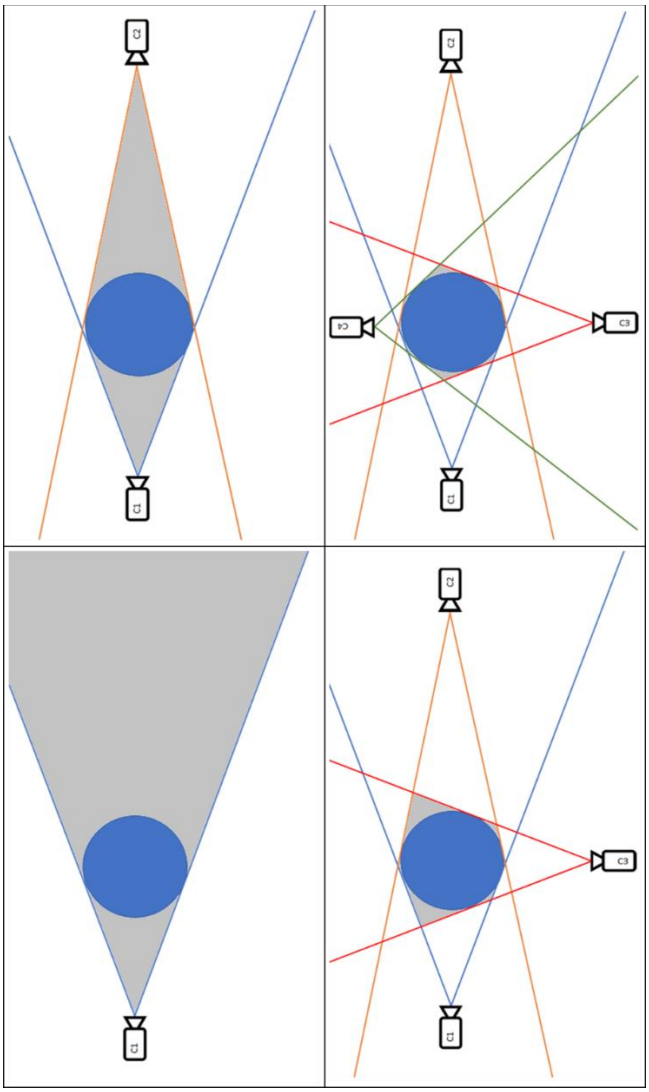



**Figure 28: voxel carving principle [81]**

The same principle can be used when there are more operators active in the work station. In that case, the visual hull will contain two separate point clouds, which can be distinguished from each other through an additional clustering step. This is shown in Figure 29.

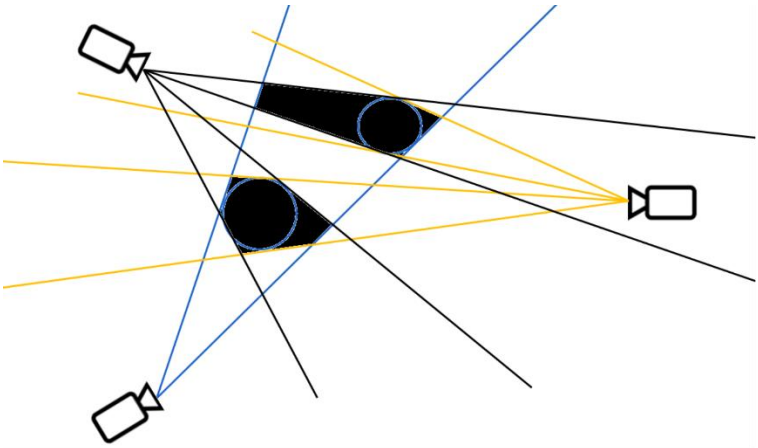


Figure 29: Visual hull in multi-operator work stations

To get accurate models, all cameras should have a full view on the object under investigation. In industrial settings there is a high risk that objects in the work station partially block this view. For example, an assembly table can completely occlude the view of one camera on the operators legs. Because of this occlusion, the operators' legs will not be included in the final model. Therefore, compensation for occluding objects needs to be taken into account.

### 2.2.2. Compensating occlusions

Occlusion is one of the main difficulties encountered when using camera systems in industrial environments. Static and moving objects can block the view of one or more cameras on the object under observation. When using the voxel carving algorithm, a partially blocked view on the operator leads to incomplete point cloud models of the operator. To overcome this problem, a self-learning algorithm is implemented to build occlusion maps for each camera in the system. These occlusion maps are later on used to determine which viewpoints should be taken into account when reconstructing the 3D model and which viewpoints should be discarded, and this for every voxel in the scene [82]. In general, three cameras are considered the minimum to generate an accurate model of an object in the three dimensional space. Because the occlusion algorithm sometimes omits

viewpoints, we use two extra cameras to safeguard the quality of the generated 3D-models.

To create this occlusion map for one camera $C_0$, first the visual hull is calculated for that one camera. This visual hull is actually the generalized infinity cone with camera as the apex of the cone and the operator's body projection as the base. Then, three cameras, $C_a$, $C_b$ and $C_c$, are randomly selected and this visual hull is projected on the view plane of these cameras, leading to the so-called reprojection masks $mask_{repr,Ca}$ , $mask_{repr,Cb}$ and $mask_{repr,Cc}$. These reprojected masks are the compared to the foreground-background masks of these cameras ($mask_{fgbg,Cx}$, x = a, b or c) to create the occlusion masks ($mask_{occl,Cx}$) which describe the invisible parts of the scene for $C_0$ in the view of camera $C_x$. These occlusion masks are subsequently used to create a new visual hull which describes the occluded voxels for camera $C_0$. This is done for every camera in the system. These occlusion maps are then used to determine which voxels in the viewpoint of the camera should be taken into account when reconstructing the 3D model of the operator. This algorithm is further explained in Figure 30. Examples of the generated masks are provided for camera $C_a$ in Figure 31.

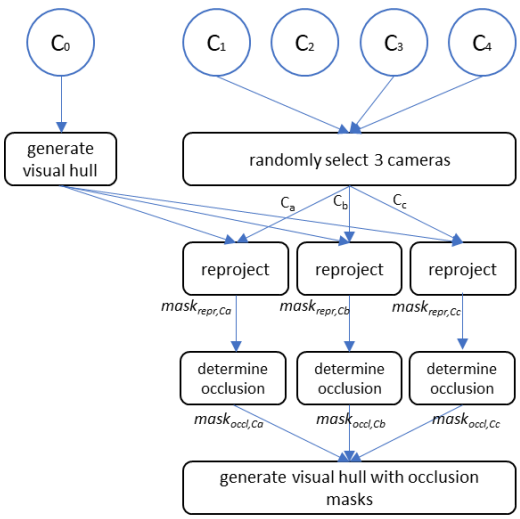



**Figure 30: outline of the occlusion removal algorithm**

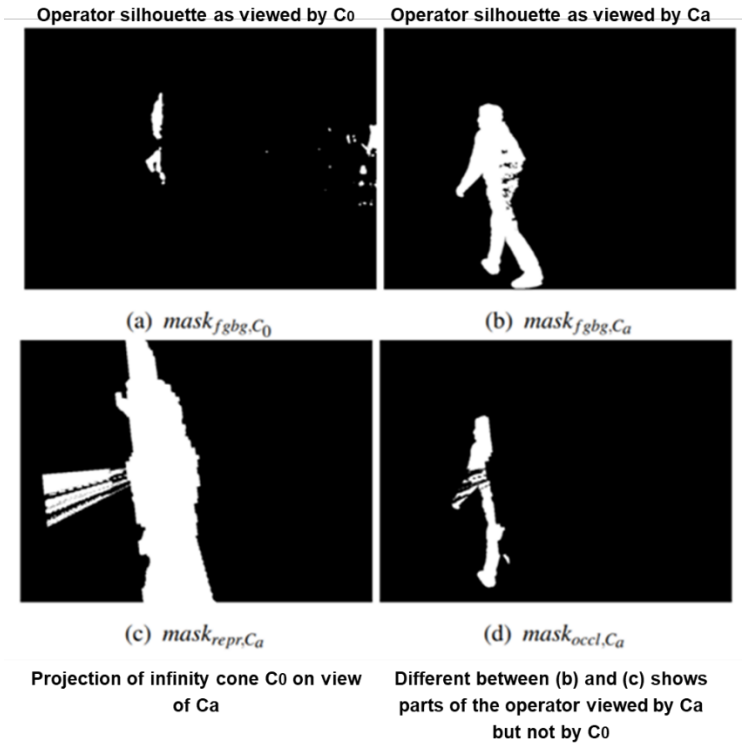



Figure 31: occlusion removal example

### 2.2.3. Vision System Output

The visual hull algorithm results in a 3D model of the operator for every frame in the video sequence. An example of such a model, is shown in Figure 32.

**Figure 32: Output of the vision system**

By calculating the models' center of gravity and projecting it on to the ground plane, we determine the location of the operator in every frame of the video stream. By doing this, we are able to track the movement of the operator in the workstation. An example of such a data stream is visualized in Figure 33. In principle, also the z-coordinate of the operators' center of mass can be calculated. However, this height data provides no real added information (no clear patterns can be identified) and therefore the z-coordinate is ignored in the remainder of this PhD thesis.

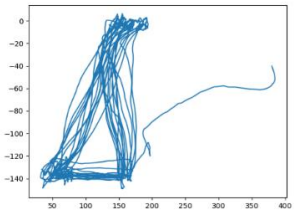



**Figure 33: Trajectories calculated based on vision system output**

## 2.3. Data Processing Framework

The output of the vision system only consists of one long stream of sampled data points describing the location of the operator in each frame of the video footage. Due to illumination changes and occlusions that are not compensated for, this data stream can contain some noise, which might negatively influence the accuracy of the classification method and the performance measurements. Also, the output of the vision system provides no information about different work cycles or tasks performed by the operator. This information is added to the data set by subdividing the data stream in useful segments and by adding semantic information to the trajectories.

In this section we describe how the data is processed before being used for outlier detection and performance measurement. The data processing framework contains three different steps: first there is a preprocessing step, afterwards the data is translated into tasks performed by the operator before being divided in different segments which describe different work cycles or tasks. This framework for data processing is visualized in Figure 34.

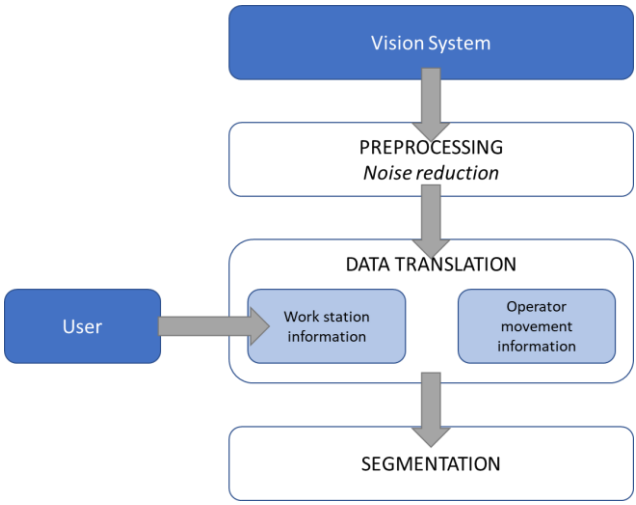



**Figure 34: Data processing framework**

In the remainder of this section, each of these data processing steps is discussed in detail.

### 2.3.1. Noise reduction

Due to disturbances such as illumination changes or remaining occlusions, the resulting trajectories might contain some noise. The calculated location points of the operator are therefore randomly scattered around his real location in the work station. This noise leads to errors on the calculation of various measures such as the travelling distance or speed of the operator. To eliminate these inaccuracies, we first need to filter the raw data to smoothen out the noise.

A Gaussian kernel [83] smoothing filter was used to cancel the noise in the trajectory data. It is widely used in image processing to remove noise and or detail in pictures. Gaussian kernel smoothing can be compared to a moving average filter. The output of these methods is obtained through a convolution of the input signal with a predefined function, the so-called kernel function. Moving average filters make use of box-shaped kernel functions, where in the case of Gaussian smoothing, a bell-shaped curve is used. The main advantage of the Gaussian kernel function is that it increases the weight of the most nearby points in the trajectory and attaches less importance to the data points that are more remote in time.

For every point p(x, y) in the trajectory, the smoothed value at time t $\tilde{p}(t)$ is calculated as follows:

$$\tilde{p}(t) = \left(\frac{\sum_j \left(w(t_j) * x(t_j)\right)}{\sum_j w(t_j)}, \frac{\sum_j \left(w(t_j) * y(t_j)\right)}{\sum_j w(t_j)}\right) \tag{3}$$

where $p_j(x(t_j), y(t_j))$ is the measured location at time j and $w(t_j)$ the Gaussian kernel function:

$$w(t_j) = e^{-\frac{(t-t_j)^2}{2\,\sigma^2}} \tag{4}$$

In this calculation, σ determines the bandwidth of the kernel function, which is in this case set to 1 second. Figure 35 shows how the Gaussian smoothing filter cancels the noise in the raw data.

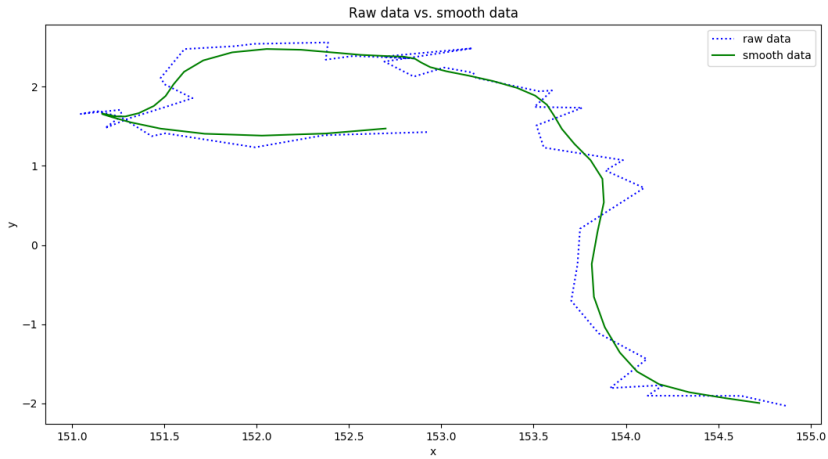



**Figure 35: Results of noise reduction method**

### 2.3.2. Data Translation

The trajectory data coming from the vision system only consists of a collection of time-stamped sample points, but they do not carry any information about the operators' tasks or performance. In order to provide useful information, the trajectory data needs to enriched with information, explaining what is actually observed in the video images. Therefore we use a framework, based on the model proposed by Alvares [84], to enrich the trajectory data with semantic information. This framework defines stops and moves in GPS trajectory data by first defining candidate stop locations, for example a museum. These candidate stop locations are described by a polygon and the object (person) is considered to have visited this location, of his/her stay time in that polygon exceeds a predefined threshold duration.

Although claimed to be generic enough for use in many applications, there are some issues when this model is applied on the specific problem of assembly operator trajectories. In our application, zones would be very close to each other and stay times are very short, which makes it nearly impossible to determine accurate threshold durations for specific zones (f.e. picking rack).

To solve this issue, the model proposed by Alvares was adapted. In our framework, stops are defined when the velocity of the operator drops below a user-specified threshold value for a prolonged period of time. These stops can happen anywhere in the work station, however they are only considered to be relevant if a stop happens within reaching distance of a so-called Point-of-Interest (PoI) in the work station layout. Stops that are outside reaching distance of a specific POI are considered to be disturbances.

To determine the location of those POI's, a heatmap of the resulting trajectory is generated. This heatmap visualizes how frequent specific locations are visited by the operator. This heatmap is generated by creating a square grid for the whole scene and calculating for each grid cell the number of frames in which the operator is located in that cell. Such a heatmap Is shown in Figure 36.

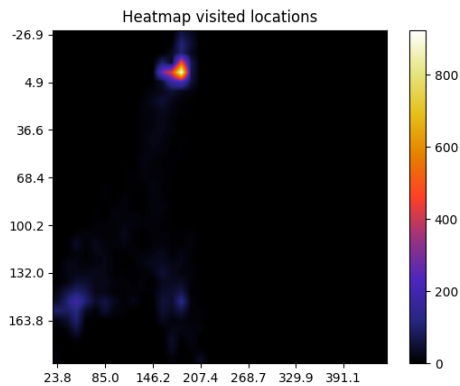



**Figure 36: Heatmap of visited locations**

This heatmap is subsequently used to determine the location of the points of interest (PoI) in the work station. Points of interests are characterized by the fact that operators spend a significant amount of their time at those locations. Their location is determined by finding the local maxima in the heatmap as shown in Figure 37.

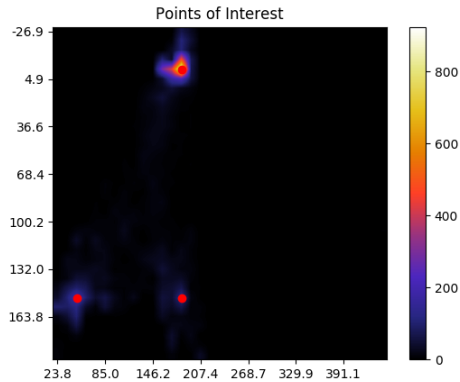



**Figure 37: Calculation of PoI based on heatmap**

These locations can be linked to typical actions that happen in an assembly work station. Typical actions in assembly work stations are parts picking, assembling parts on the final product or reading instructions. This way an annotated map of the work station is generated.

In some cases, the necessary information about the work station is readily available in the form of, for instance, CAD drawings. In that case, the location of the PoI's can be matched to the information in the CAD files to automate the generation of this annotated map. If this information does not exist, the calculated locations of the PoI's are presented to the methods engineer. Based on his knowledge of the work station or the video images, he is able to link specific locations to certain actions.

Two different types of actions are defined: stops and moves. Events are defined when a stop occurs in the neighborhood of one of the points-of-interest in the work station layout. The description on the PoI where the event happens, defines what task is performed during in this event. For each event, the location and duration are saved to an event list file.

When the operator has to move from one PoI to another in between events, moves are registered. These moves are also added to the event list. This way the event list describes the trajectories and all

actions performed in a human-readable manner. An excerpt from such an event list is provided in Figure 38. Besides the description of the tasks performed, crucial information such as task times is logged in the event list.

```
---------------------------------------------------------------------------------------------------------------------------
POI - category - location - duration (ms)
move - start location - destination - travel distance - average speed - duration
['moving', 'assembly', 'Rack_3', 1.1316544537437296, 0.8827263535020522, 1300]
['Rack_3', 'Picking', [152.34858362128202, -149.26521728234806], 4900]
['moving', 'Rack_2', 'Rack_1', 0.583826270206011, 0.96665976372047047, 650]
['Rack_1', 'Picking', [69.855200142233599, -131.25345220250145], 4150]
['moving', 'Rack_1', 'assembly', 1.1052905603557148, 1.4480360922646605, 800]
['assembly station', 'Assembly', [124.44227923867818, -16.006946940062374], 8200]
---------------------------------------------------------------------------------------------------------------------------
```

**Figure 38: Excerpt from the automatically generated eventlist**

### 2.3.3. Trajectory segmentation

The vision systems provides one single stream of time-stamped xy-coordinates of the operators position. If we want to use this information to evaluate the operators' performance, this data stream needs to be divided into segments. These segments could be a complete work cycle or a specific task, depending on the level of detail required for the analysis. Sometimes this segmentation can be done by synchronizing the vision systems' output to information coming from the MES system. This information can indicate when a specific product is finished and when a new work cycle starts.

However, this information is not always available. Performing the segmentation manually, on the other hand, would be time-consuming. Therefore, the segmentation of the resulting trajectory in interesting segments, has been automated.

To do this, the resulting trajectory is linked to the information in the annotated work station map, generated earlier. More specific, we define an assembly zone around the location of the end product. Typical assembly actions are structured as follows: first a part is picked from a kitting cart or rack, then it is moved towards the assembly zone, subsequently it is positioned on the final assembly before ultimately being fixed to the end product. Under the assumption that

each assembly task or work cycle ends with an action on the final assembly, we can define a new work cycle when the operator leaves the assembly zone. The segmentation could also be performed based on the event list previously described. However we opted to start from the raw trajectory data so we could use the framework purely for outlier detection without the need to define completely annotated work station layouts.

Due to disturbances and noise around the edges of the assembly zone, this segmentation method may sometimes result in unrealistically short segments. When the duration of a particular segment is significantly less than the average duration of a work cycle, it is considered to be a noisy result of the segmentation procedure and this cycle is added to the previous work cycle. This filtering method is purely based on the duration of the resulting segments. Therefore it does not guarantee that small corrective actions, such as going back to check if there are no mistakes, are filtered before classification. However, since the nature of these corrective cycles will strongly differ from normal work cycles, the classification method will eventually treat them as outliers, which are returned to the analyst for further investigation.

Figure 39 shows how the trajectory data set visualized in Figure 33 is subdivided in segments describing different work cycles.

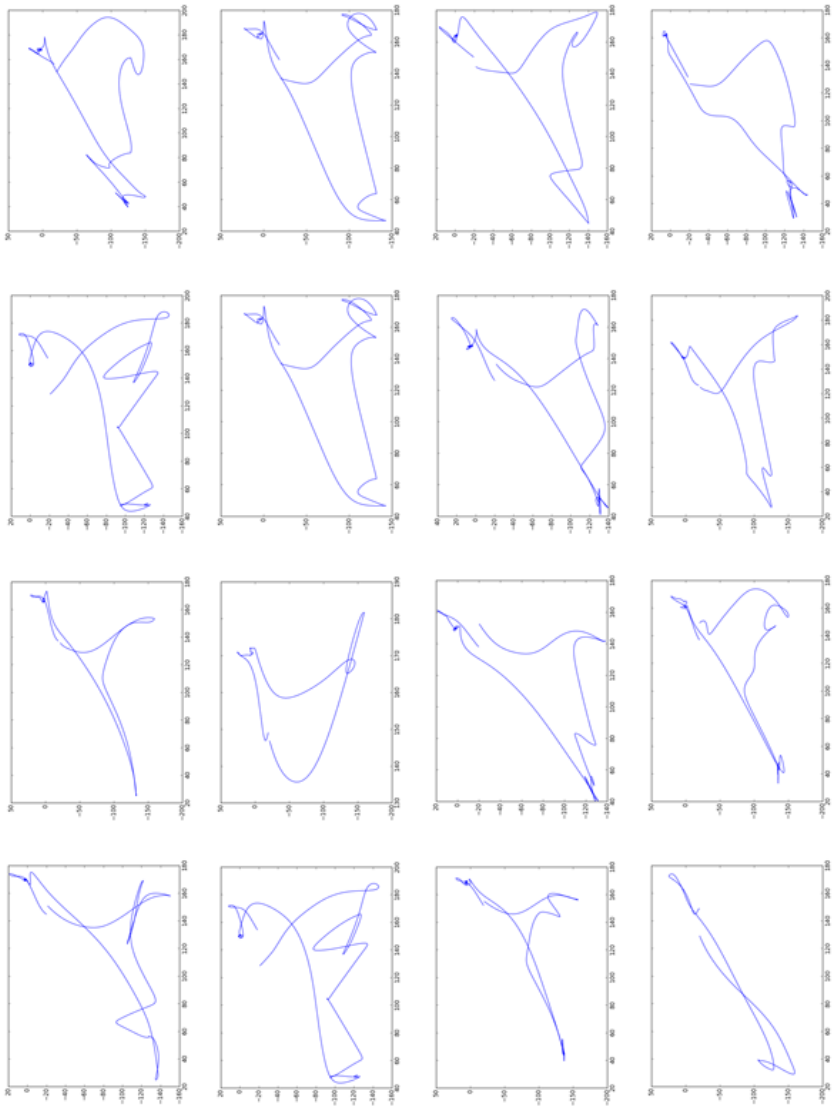


**Figure 39: Segmentation of the trajectory**

## 2.4. Conclusions

Camera systems offer three advantages compared to other Real-Time location systems when monitoring assembly line operators: (1) cameras are less intrusive than wearable sensors, cameras have become relatively inexpensive and (2) the resulting video images contain large amounts of useful information which can be used for the analysis of the work stations' performance. Therefore, we propose a multi-camera based operator monitoring system in this chapter.

Throughout this research, three different data sets have been used. In the first section of this chapter, these data sets are described in more detail. The first data set was created by monitoring the movements of human operators in a simulated assembly line work station in a laboratory setting. The second data set was found in literature and contains a higher number of trajectories, which is useful to validate the classification framework described in the next chapter. The final data set was created by recording the behavior of an assembly line operator in a real assembly work station at a supplier of subassemblies for the automotive industry.

In the second section, a multi-camera vision system to monitor the behavior of human operators was introduced. Through the use of the voxel carving principle, a 3D reconstruction of the operators' body is generated for every time frame in the video images. Thereby, special attention was given to the issue of occluding objects in the view of the cameras. This 3D reconstruction is then used to monitor the operators' movement through the work station. The presented video processing algorithms were developed by researchers of IMEC and improved during the joint StarTT project Complexity [77] for use in industrial environments.

The trajectories returned by this vision system contain noise and lack the necessary information to explain the operators' behavior. Therefore a data processing framework, based on the layout of the work station, is presented in the third section of this chapter. This framework results in an automated transcription of the events that took place in the recorded video images. This so-called event list can

be used to facilitate the generation of work instructions and provides automated time measurements of all tasks performed.

# Chapter 3 Operator Trajectory Clustering

With the development of new location based positioning devices, there is an increasing trend to monitor and capture the trajectories of moving objects. The captured data potentially contains a lot of valuable information. This has led to an increasing interest of researchers in trajectory clustering methods and algorithms. Trajectory clustering is an efficient method to group similar objects, in this case trajectories, in clusters, thereby finding the underlying structure in a large unstructured data set of non-categorized objects.

In this chapter, a method to automate the analysis of the work performed in assembly line work stations is developed. The method automatically clusters the trajectories followed by the operators during his work cycle and differentiates between normal and irregular patterns. There are two main advantages to clustering human operator trajectories: (1) Irregular trajectories indicate a source of variability in the process. This kind of variability in a manufacturing process is not desirable and is often caused by all sorts of different problems. These trajectories are worth further investigation. The classification method speeds up the analysis process by filtering the trajectories and only feeding back relevant events to the analyst. (2) Most time study techniques rely on statistical methods. In order to obtain significant results, a large amount of measurements needs to be done. The method presented in this chapter, supports the analyst in this time-consuming task by grouping the similar trajectories and automatically performing the time measurement. This way, the operator can be monitored over a longer period of time without the need for manual measurements. This provides more accurate and reliable data with less effort.

This chapter provides an overview of existing clustering methods and algorithms described in literature. Based on this literature review, a

suitable method for clustering human operator trajectories in an industrial setting is presented. Finally this method is validated based on a number of experimental datasets and compared to commonly used alternative methods.

## 3.1. Literature Review

The rapid development of GPS devices, sensor networks, wireless communication technology and video analysis techniques has made it possible to capture massive amounts of trajectory data of all kinds of moving objects. On top of that, cloud computing and storage technology provides practically unlimited data storage capacity. This results in an ever increasing amount of trajectory data of all sorts to be stored captured and stored in massive databases. These data sets potentially contain large amounts of valuable information. To unveil this information, efficient and effective methods and algorithms to analyze this data are required. Therefore, trajectory clustering methods have drawn a lot of attention of researchers in the past couple of years. Clustering methods group data instances into subsets in such a manner that similar instances are grouped together, while different instances belong to different clusters [85]. Clustering methods unveil the underlying structure of a data set in an efficient manner. Formally, the clustering structure of a data set S can be described as a set of subsets C = $C_1$, $C_2$, …, $C_k$, such that: $S = \bigcup_{i=1}^{k} C_i$ and $C_i \cap C_j = \emptyset$ for $i \neq j$. This means that each instance of the data set S belongs to exactly one single subset. Clustering methods also produce an interesting by-product, namely outliers. These are single-instance subsets of the data set. Therefore, outliers or anomalies are trajectories that show little or no similarity to the other trajectories in the data set [86]. In an industrial context, these anomalies are considered to be interesting for further investigation, since they may indicate irregularities or problems in the concerning process.

### 3.1.1. (Dis)similarity measures for trajectories of moving objects

An important factor in the success of clustering algorithms is the choice for a suitable distance or similarity measure. In traditional clustering methods, the distance between data points is unambiguously determined by an intuitive distance measure, such as

Euclidean or Manhattan distances. Trajectory data however are time series which are often multi-dimensional in nature and in most cases have different lengths. Traditional distance measures are not easily applicable to this type data. Therefore, it is important to select a measure which can comprehensively compare the similarity and differences between two trajectories. With the advent of novel tracking and recording technologies and the related increase of available trajectory data, there is a growing interest in literature in new, more suitable distance measures for trajectory data. This section provides an overview of existing similarity measures and their applicability to the trajectory clustering problem.

***Euclidean distance***

One of the most intuitive distance measures for clustering trajectory data is the Euclidean distance. It was applied by Fu [87] on the problem of anomaly detection in the trajectories of vehicles. Given two p-dimensional trajectories $L_i$ ($a_1$, $a_2$, ..., $a_n$) and $L_j$ ($b_1$, $b_2$, ..., $b_n$) with length n, the Euclidean distance can be calculated as follows[86]:

$$D(L_i, L_j) = \frac{1}{n} \sum_{k=1}^{n} \sqrt{\sum_{m=1}^{p} (a_k^m - b_k^m)^2} \qquad (5)$$

There are two main advantages to evaluating the similarity of two trajectories using the Euclidean distance: (1) Euclidean distance is parameter-free and (2) the complexity of the algorithm is linear which means that it can be applied to larger data sets. Despite the clear advantages, there exist only a limited number of successful applications of the method to the trajectory clustering problem. Reasons are the sensitivity of the measure to noise in the dataset, the inability to cope with time deformations of the trajectories and the requirement that the length of both trajectory segments needs to be equal.

Bashir [88] used a combination of principle component analysis (PCA) and Euclidean distance to reduce the dimensions of the data. The trajectory was first represented as a 1-dimensional signal by concatenating the x and y projections before being converted into the most significant PCA coefficients. Trajectory similarity is subsequently

calculated as the Euclidean distance calculated with PCA coefficients. The reduction in dimensions leads to a faster method and an increased ability to cope with noise in comparison with the application of Euclidean distance on the raw data. However, the requirement for trajectories of equal length remains, which limits the practical use of the method.

***Hausdorff distance***

The Hausdorff distance is a similarity measure commonly used in computer vision [89]. More recently it has been successfully applied on the problem of trajectory clustering [90, 91]. The Hausdorff distance $D_h$ is defined as follows:

$$D_h(L_i, L_j) = \max(h(L_i, L_j), h(L_j, L_i)) \tag{6}$$

$$h(L_i, L_j) = \max_{a \in L_i}(\min_{b \in L_j}(dist(a, b)) \tag{7}$$

In the case of trajectory data, dist(a, b) is the Euclidean distance between a and b belonging to $L_i$ and $L_j$ respectively.
The Hausdorff distance can intuitively be explained as the greatest of distances between a sampling point in the first trajectory to the closest point in the second trajectory. In practice this means that two trajectories are close in the Hausdorff distance, if every point of the first trajectory is close to some point of the second trajectory.
Despite its successful application in a number of applications, the Hausdorff distance has two major shortcomings. Since the Haussdorf distance is shape-based, it doesn't take into account time deformations of the trajectory. Also, looking at the intuitive definition of the measure, it is clear that the Hausdorff distance is very sensitive to outliers in the trajectory data. One deviating point in the trajectory data will yield a poor similarity score between two trajectories, even if most parts of the two trajectories are very similar.

***One way distance***

Shape-based similarity measures, such as the Hausdorff distance, tend to be rather sensitive to noise and outliers in the trajectory data. Also, most existing similarity measures that are commonly used for

comparing trajectory data, require a mapping in which the order of data points is preserved at all times. For example: assume $a_1$ and $a_2$ are two points of trajectory $L_1$ and $b_1$ and $b_2$ are two points of trajectory $L_2$. This would mean that, if $b_1$ is mapped on to $a_1$, $b_2$ is mapped on to $a_2$ and if $a_1$ appears in $L_1$ before $a_2$, then $b_1$ is required to appear in $L_2$ before $b_2$.
In some cases, this is a limitation to the similarity measurement. To overcome these problems, Lin and Su [92] developed a similarity measure based on the one-way distance (OWD).

Assume $D_{point}(a, L_2)$ represents the distance between point a and trajectory $L_2$:

$$D_{point}(a, L_2) = \min_{b \in L_2} (D_{euclidean}(a, b)) \tag{8}$$

In that case, we can define the one-way distance between trajectory $L_1$ and trajectory $L_2$ as follows:

$$D_{OWD}(L_1, L_2) = \frac{1}{|L_1|} \sum_{a \in L_1} D_{point}(a, L_2) \tag{9}$$

The one-way distance is not symmetrical, therefore the distance between two trajectories is calculated by taking the average of their respective one way distances:

$$D = \frac{1}{2} * (D_{OWD}(L_1, L_2) + D_{OWD}(L_2, L_1)) \tag{10}$$

***Longest common subsequence***

Vlachos, Buzan and Liu [93-95] used the longest common subsequence (LCSS) measure for the classification of 2D trajectory data to overcome issues other distance measures have with noisy data. LCSS is a similarity measure that finds the alignment between two trajectories that maximizes the length of the common subsequence and can be recursively calculated as follows:

$$\begin{cases} 0, & if\ L_1\ or\ L_2 = \emptyset \\ 1 + LCSS_{\varepsilon,\delta}(HEAD(L_1), HEAD(L_2)), & if\ \left\|a_{x,n} - b_{x,m}\right\| < \varepsilon, \left\|a_{y,n} - b_{y,m}\right\| < \varepsilon\ and\ \|n - m\| < \delta \\ MAX\left(LCSS_{\varepsilon,\delta}(HEAD(L_1), L_2), LCSS_{\varepsilon,\delta}(L_1, HEAD(L_2))\right), & otherwise \end{cases} \tag{11}$$

Where $L_1\ and\ L_2$ are two trajectories respective lengths n and m and where HEAD($L_1$) represents the subsequence $((a_{x,1}, a_{y,1}), \dots, (a_{x,n-1}, a_{y,n-1}))$ of subsequence $L_1 = ((a_{x,1},$

$a_{y,1}), \dots, (a_{x,n}, a_{y,n}))$. LCSS is controlled by two parameters: $\delta$ limits how far one can go to match a given point of one trajectory to a point in the other trajectory. The matching threshold $\varepsilon$ determines whether two points in two trajectories are close enough to be considered as a match. Based on the longest common subsequence, the similarity of two trajectories can be calculated as follows:

$$S_{LCSS}(L_1, L_2, \varepsilon, \partial) = \frac{LCSS_{\varepsilon,\delta}(L_1, L_2)}{\min(n, m)} \tag{12}$$

Most clustering and classification methods require a distance measure instead of a measure of trajectory similarity. Therefore, a distance measure based on the LCSS of two trajectories can be computed according to Vlachos [93] using following formula:

$$dist_{LCSS}(L_1, L_2, \varepsilon, \partial) = 1 - S_{LCSS}(L_1, L_2, \varepsilon, \partial) \tag{13}$$

The LCSS distance measure has been effectively and efficiently applied to a wide range of different trajectory data classification problems. One of the main advantages of LCSS lies in the fact that it allows for unmatched points in the trajectories under comparison. This way, the effect of erroneous data points or noise in the trajectory data is limited. However, LCSS heavily relies on two user-defined parameters. Choosing the optimal values for these parameters remains a challenging problem which can have a severe impact on the results of the clustering method.

***Fréchet distance***

Another commonly used distance measure to compare two trajectories was introduced by Maurice Fréchet. It is often referred to as the “dog-leash distance” because it can intuitively be explained by imagining a person walking its dog on a leash. If the two of them are each walking on two separate paths without backtracking from one endpoint to another, then the Fréchet distance can be represented by the minimum length of the leash required to connect the dog and its owner [96]. Formally, the discrete Fréchet distance can be described as follows:

Given two trajectories $L_i$ ($a_1$, …,$a_n$) and $L_j$($b_1$, …, $b_m$) with respective lengths n and m, we can define a coupling C (($a_{u1}$, $b_{v1}$), …, ($a_{ul}$, $b_{ul}$)) between distinct pairs of $L_i$ and $L_j$ such that $u_1$ = , $v_1$ = 1, $u_l$ = n, $v_l$ = m, and for all l = 1,…, l, $v_{i+1} = v_i$ or $v_{i+1} = v_i+1$ and $u_{i+1} = u_i$ or $u_{i+1} = u_i+1$. The length ||C|| of such a coupling C is the length of the longest link in C:

$$||C|| = \max_{i=1,\dots,l}(d(a_{ui}, b_{vi})) \quad (14)$$

Where d($a_{ui}$, $b_{vi}$) represents the Euclidean distance between points $a_{ui}$ and $b_{vi}$. The Fréchet distance between two given polygonal trajectories $L_i$ and $L_j$ is defined to be [97]:

$$D_f(L_i, L_j) = \min(||C||, where\ C\ is\ a\ coupling\ between\ L_i\ and\ L_j) \quad (15)$$

Different from the distance measures previously mentioned, the Fréchet distance takes into account the sequential relationships of the points in the trajectories while measuring their similarity. This typically results in a more valuable similarity estimation than measures that are purely shape-based. This is visualized in Figure 40, where H indicates the Hausdorff distance and F represents the Fréchet distance. However both trajectories are geometrically close to each other and even share common points, they are in reality very different. This is indicated by a low Hausdorff distance compared to a very high Fréchet distance estimation.

***Dynamic time warping***

Dynamic time warping is originally developed to compare sound signals for speech recognition purposes. By now, It has been successfully applied on various types of time-dependent data in order to cope with time deformations of time series [98].

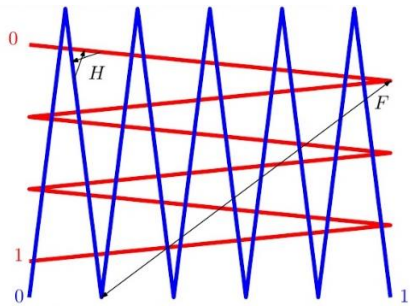



**Figure 40: Frechet vs. Haussdorf**

Dynamic time warping makes it possible to compare time series of different lengths. The warping method aims to minimize the distance between two time series by stretching them in a well thought out manner. Originally, the DTW algorithm was developed for one-dimensional time series. However the method can easily be adapted to cope with multi-dimensional time series, such as trajectory data.

There are numerous examples of the application of DTW on all sorts of trajectory data. Vaughan [99] used this technique to assess the skill level of aspiring physicians by comparing trajectories obtained through a virtual reality haptic simulator for epidural needle insertion. Cheng [100] proposed a framework to identify paths with a high probability of traffic congestion by using DTW to compare the similarity of trajectories in intelligent transportation systems. Successful applications of the warping algorithm on human motion trajectories, however not in an industrial setting, can be found in the work presented by Pohl [101] and Blackburn [102].

Given two time series $L_1 := (a_1, a_2, \dots, a_N)$ and $L_2 := (b_1, b_2, \dots, b_M)$ with respective lengths $N, M \in \mathbb{N}$, DTW tries to find an optimal warping path that minimizes the distance between two time-dependent sequences. The warping path p is a sequence of points, $p = (p_1, p_2, \dots, p_K)$ with $p_k = (n_k, m_k) \in [1:N] \times [1:M]$ for $k \in [1:K]$ under three conditions:

- **Boundary condition**: $p_1 = (1, 1)$ and $p_k = (N, M)$. The warping path starts and ends at the respective beginning and end of both sequences.

- **Monotonicity condition**: $n_1 \leq n_2 \leq \ldots \leq n_K$ and $m_1 \leq m_2 \leq \ldots \leq m_k$. The warping path preserves the time sequence in trajectory matching. In other words, "Backtracking" is not allowed.
- **Continuity condition**: $p_{k+1} - p_k \in \{(1, 0), (0, 1), (1, 1)\}$ for $k \in [1:K-1]$. All points of both time-dependent sequences are included in the warping path.

To find the optimal alignment path, an n by m cost matric C is constructed recursively as follows [103]

$$C(Xi, Yj) = \delta(xi, yj) + \min\{ C(Xi-1, Yj-1), C(Xi-1, Yj), C(Xi, Yj-1) \} \quad (16)$$

where $X_i$ an $Y_j$ are the respective subsequences ($x_1$, $x_2$, … , $x_i$) and ($y_1$, $y_2$, …, $y_j$) and $\delta(x_i, y_j)$ is the Euclidean distance between $x_i$ and $y_j$. The DTW distance between two time series X and Y based on their optimal warping path, can be calculated using the equation below:

$$DTW(X, Y) = \frac{C(X_n, Y_m)}{K} \quad (17)$$

where K = m + n. This weighing factor is used to normalize the length difference of different warping paths [104]. Figure 41 and Figure 42 demonstrate how two one-dimensional time series, indicated by the dotted and triangular markers, are aligned using the DTW algorithm. Points that are matched by the algorithm are connected by the dashed lines. The time-warped distance between the two time series is calculated by summing up the difference in value for all matched data

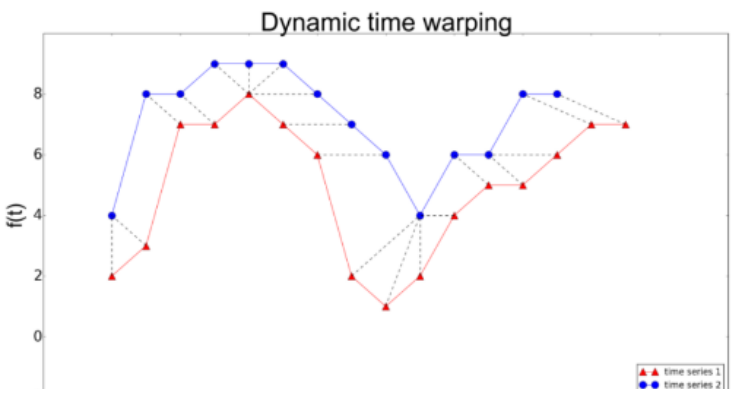



**Figure 41: DTW alignment example**

points. In case of the two-dimensional paths, the time-warped distance is equal to the sum of the sum of the Euclidean distance between matched data points.

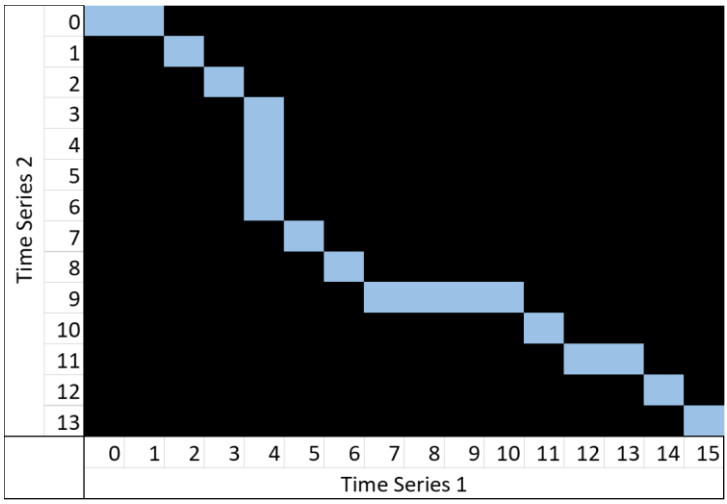



**Figure 42: DTW Warping path**

The pseudocode of the DTW algorithm is provided in Figure 43.

```
DTWDist(s: [X: (x1, y1), ..., (xn, yn)], Y: [(x1, y1), ..., (xm, ym)])

1.  Initialization
2.  for i:= 1 to n
3.      C[i, 0] := infinity
4.  for i := 1 to m
5.      C[0, i] := infinity
6.  C[0, 0] := 0
7.  Calculating DTW distance
8.  for i := 1 to n
9.      for j := 1 to m
10.         cost := d(Xi, Yj)
11.         C[i, j] := cost + min( C[i-1, j], C[i, j-1], C[i-1, j-1])
12. k := m + n
13. return C[n, m]/K
```

**Figure 43: DTW algorithm outline**

***Similarity measures: summary***

In this section, an overview of commonly used similarity measures for moving object trajectories is provided. While the Euclidean distance is the most intuitive measure, it is not capable of handling trajectories of different length. Therefore, this measure will not be investigated further in this chapter. The fit-for-purpose of the other measures is investigated experimentally later on in this chapter. The advantages and disadvantages of these measures are summarized in Table 5.

**Table 5: Similarity measures overview**

| | parameter independent | time deformations | noise sensitivity | calculation time |
|---|---|---|---|---|
| **euclidean** | ✓ | × | | O(n) |
| **haussdorf** | ✓ | × | -- | O(n+m) |
| **OWD** | ✓ | × | - | 0(n+m) |
| **LCSS** | × | × | + | O(n*m) |
| **Fréchet** | ✓ | × | - | O(n*m) |
| **DTW** | ✓ | ✓ | ++ | O(n*m) |

In many clustering applications, a normalization step is performed first. According to recent literature concerned with time series clustering, z-normalization is an essential preprocessing step in structural pattern mining [105] because it allows a mining algorithm to focus on the structural similarities/dissimilarities rather than on the amplitude-driven ones. However, in our case, amplitude is important as we are interested in the absolute location of the operator. In Figure 44 we show how valuable information about the operators' position could get lost through normalization. Therefore, we use non-normalized trajectories as input for the clustering algorithm.

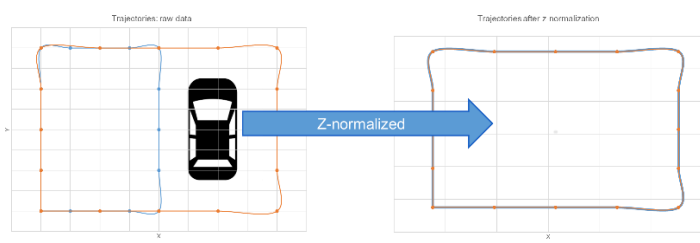



**Figure 44: Effect of z-normalization of trajectory data**

### 3.1.2. Clustering methods for trajectories of moving objects

A wide range of different clustering methods and algorithms have been proposed in literature. Traditionally, these clustering methods are subdivided into two main categories, namely hierarchical and partitional clustering methods [106]. According to Han [107] three more categories can be identified: density-based, model-based and grid-based clustering methods. Each of these methods are based on different principles and therefore have their own preferred application domain. Below, the basic principles of each of these categories of clustering methods that are commonly used for trajectory clustering, are described together with their advantages and drawbacks when applied to this domain.

***Partitioning-based methods***

Partitioning-based methods start from an initial (random) partitioning of the objects in the data set and try to minimize a given criterion by iteratively relocating objects between k clusters until a (local) optimum is achieved [85]. These methods require only one parameter k ($k \leq n$, where n is the number of objects in the data set), which is mostly predefined by the user. Typical examples of partitioning-based methods are k-means [108] and k-medoids clustering [109]. The convergence of typical partitioning-based methods is local and therefore a global optimum cannot be guaranteed. Theoretically, this problem of local minima could be overcome be using exhaustive enumeration since the number of objects in the data set and the number of clusters are both finite numbers. In practice however, finding a globally optimal solution for this problem is proven to be NP-hard and therefore not useful in practice. The number of possible partitions for n observations into K clusters is a Stirling number of the second kind and can thus be calculated as follows:

$$S_n^K = \frac{1}{K!} \sum_{i=0}^{K} -1^{k-i} \binom{K}{i} i^n \tag{18}$$

As an example, a small data set containing 20 objects and 3 possible clusters would require more than 580 million possible solutions to be evaluated. This shows that exhaustive enumeration of all possible partitions is practically unfeasible, even for relatively small datasets and a predefined number of clusters [110]. If the number of clusters would be unknown, the problem would become even more extensive.

Partitioning methods such as k-means and k-medoids are often used because of their speed of convergence towards a local optimum, their ease of implementation and interpretation of the results and their adaptability to sparse data [111]. However there exist a number of disadvantages to these methods which render them difficult to apply on human motion trajectory data: (1) the number of clusters needs to be known beforehand, requiring a priori knowledge which is unavailable in most cases, (2) the choice of the initial partitioning largely determines whether a local or global optimum is reached, (3) these methods are very sensitive to outliers and noisy data [85] and (4) partitioning and relocation methods require that the complete data set is loaded into the memory, leading to problems of memory cost when dealing with larger data sets [112].

***Hierarchical methods***

Hierarchical clustering methods aim to structure a given data set by building a hierarchy of clusters. These methods can either follow a bottom-up or top-down approach. The first case is referred to as agglomerative hierarchical clustering (AHC) methods. These methods start by partitioning each object in the data set in its own cluster and proceeds by incrementally merging the 'most similar' clusters as it moves up the hierarchy. Divisive methods take a top-down approach on the clustering problem. In this case, all objects are considered to belong to the same cluster, which is subsequently split up as the method moves down the hierarchy [113].

Hierarchical clustering methods usually result in a dendrogram. Dendrograms are a visual representation of the nested grouping of data objects and the similarity levels at which these groupings change. An example of a dendrogram is given in Figure 45.

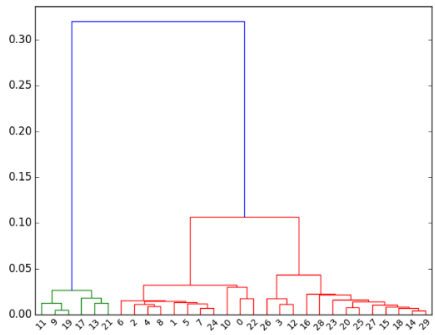



**Figure 45: dendrogram example**

As mentioned earlier in this chapter, there are several different distance measures that can be used in hierarchical clustering methods. Besides the distance measure, the decision to merge or divide certain clusters is also based on a linkage rule, which decides what criterion is optimized by the clustering algorithm. Commonly used linkage rules are:

***<u>Linkage criteria</u>***

1. **Single linkage:** In single linkage agglomerative clustering methods, the similarity between two clusters is based on the similarity of their most similar elements [114]. In other words, single linkage clustering methods combine the clusters that contain the closest pair of elements which don't belong to the same cluster. In this case, the assessment of cluster similarity is solely based on the similarity of a single pair of elements. This results in a very local linkage criterion which does not fully reflect the distribution of elements in a cluster. As a consequence, clusters that are very distant from each other may be merged because of the presence of a single pair of

elements that are close to each other. This phenomenon, called the chaining effect, is visualized in Figure 46.

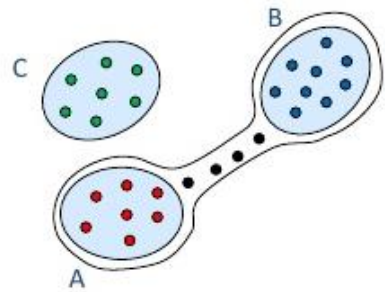



**Figure 46: Chaining effect in single linkage AHC**

2. **Complete linkage:** The complete linkage or 'furthest neighbor' criterion works in a similar way as single linkage, with the difference that this method considers the furthest distance between pairs instead of the minimum distance. This approach solves the problem of possible chaining effects, but introduces a new issue. When a cluster contains outlying data, the complete linkage criterion could prevent the merge of close-by clusters because the distance between these clusters is only determined by the outlying object. This is shown in Figure 47, where the outlying data point in the green cluster results in a large distance between the blue and green cluster [115].

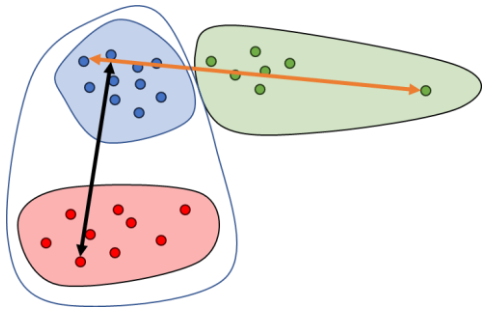


Figure 47: disadvantage of complete linkage

3. **Average linkage:** The average linkage criterion, also referred to as Unweighted Pair-Group Method using Arithmetic Averages, calculates the distance between to separate clusters as the average distance between one object in one cluster to all entities in the other cluster. This linkage criterion overcomes the shortcomings of complete and single linkage clustering methods [116].
4. **Ward's linkage method:** Ward's minimum variance method is a special case of the objective function approach presented by Ward [117]. In this approach, the decision on which clusters to merge next in the agglomerative clustering is based on the optimal value of a certain objective function. This objective function can be any function that reflects the goal the user aims to achieve. In the case of Ward's minimum variance method, the aim is to find the pair of clusters that results in the minimum increase of total within-cluster variance after merging.

Hierarchical clustering methods offer two main advantages. First of all, hierarchical clustering algorithms result in a similarity structure instead of one single partitioning of the data objects. Depending on the application, the user can choose between different partitions according to the desired similarity level. The second advantage is the versatility of hierarchical clustering methods. Hierarchical methods tend to work well with a wide variety of similarity measures and maintain good performance on data sets with non-isotropic clusters [85]. On the downside, hierarchical methods have no backtracking

capability, which means that once a merge or division is performed, it cannot be undone. Furthermore the time complexity of these methods is at least $O(m^2)$ (where m is the number of data objects in the data set), which yields rather large computational time when dealing with larger data sets.

***Alternative methods***

Most clustering methods can only deal with clusters of convex (spherical) shape. Typical *density based methods* such as DBSCAN [118] and OPTICS [119] are able to group clusters in any shape. These methods differ substantially from other clustering algorithms in the way clusters are defined. The basic idea behind density-based clustering is that for each object in a particular cluster, the neighborhood of a given radius $\varepsilon$ must contain at least a minimum of $n_{pts}$ data objects [120]. In other words: the cardinality of this neighborhood has to exceed a specified threshold value. Density-based methods tend to perform rather well on real-world applications because they are robust to problems such as noise and outliers. This can be explained by the fact that outliers usually only have a limited effect on the overall density distribution of the data set. However, these methods are heavily depending on user-defined parameters. Determining the right parameter settings for a specific case remains a difficult task.

To bypass the parameter-dependency of traditional density-based methods, [121] proposed an adaptive trajectory clustering method based on grids and density (ATCGD). ATCGD consists of three separate phases: (1) in the partitioning phase all trajectories are approximated by a set of linear subtrajectories in order to compress the trajectory data. In this case, the partitioning of trajectories is based on the average angular difference. (2) Afterwards these subtrajectories are mapped into cells of a predefined grid during the mapping phase. During this mapping procedure, the optimal values for $n_{pts}$ and $\varepsilon$ are computed and used during the clustering phase (3), using a clustering method that is based on the DBSCAN algorithm.

Similar *partition and group* algorithms have been proposed for classifying trajectory data to overcome two main shortcomings of

traditional clustering methods that consider the whole trajectory as the basic unit: (1) these methods overlook local characteristics that occur in complex moving object trajectories and (2) common sub-trajectories or local patterns can't be found [86]. Partitioning of trajectories can be performed based on a wide range of spatial and temporal criteria, such as velocity, curvature, acceleration, location and shape. Buchin [122] proposed a framework that partitions a given trajectory in a minimum number of segments based on any (combination) of these criteria. Partition and group algorithms tend to perform very well on very long and complex trajectories of high dimensionality. These algorithms use certain partitioning criteria to form basic units for the clustering algorithm. The main drawback of this concept lies in the fact that the results of the clustering method is heavily relying on the partitioning criteria used.

A number of authors focus on the reduction of the trajectory uncertainty to improve clustering results and enhance the utility of trajectory data and clustering methods. The uncertainty of trajectories means that, although objects move continuously, tracking technology only allows for location updates at discrete times. What happens to the object in between these sampling points, remains unknown and can only be estimated, creating uncertainty in the data at hand. Examples of clustering methods for uncertain trajectory data are described in [123] and [124] .

Today, the wide range of available sensors makes it possible to enrich the trajectory data, which generally only consists of a stream of timestamped locations, with additional information such as elevation, rotation, direction or acceleration. Recently, an increasing number of researchers started to focus on using this kind of semantic information in the clustering process. Palma [125] proposed a method that defines moves and stops in trajectory data. Stops are interesting locations where the object stayed for a certain amount of time. This information, when linked to geographical data, results in a semantic description of a trajectory. Such a description, for example the touristic places a person visited and the time he/she stayed at that place, can be used as a basis for clustering. Other examples of semantic trajectory data clustering can be found in [126-128].

### 3.1.3. Determining the number of clusters in a data set

Because of its capability to find a similarity structure in a data set without prior knowledge about the number of patterns (different trajectories) in that data set, hierarchical agglomerative clustering is selected as the most suitable clustering method for this application. AHC works with practically any (dis)similarity measure and its outcome is not dependent of any specific parameter settings. The results of a AHC routine are typically visualized in a so-called dendrogram. The final decision on which trajectories can be classified in the same cluster and which trajectories represent abnormal operator behavior, is taken by cutting the dendrogram at a particular height. There exist different methods to determine the optimal cutting height. In this section, four commonly used methods are explained. These four methods were implemented and their suitability for clustering human operator trajectories was investigated later on in this chapter.

***Threshold cut***

A commonly used and straightforward method to decide on the number of clusters in a data set consists of cutting the dendrogram at a predetermined height, usually expressed as fraction of the total height of the dendrogram. The threshold cut method is visualized in Figure 48.

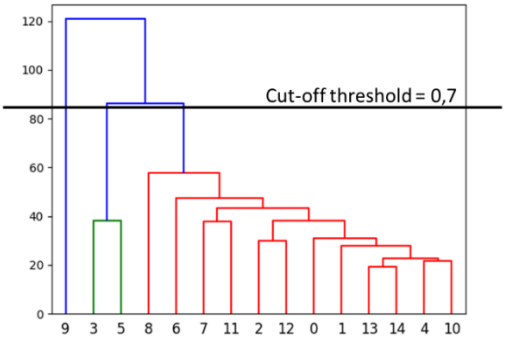



Figure 48: threshold dendrogram cut-off

The main advantage of this method lies in its simplicity, however the results are heavily depending on the chosen cut-off value. Also, the method doesn't take into account the structure of the dendrogram. For example, consider a data set that consists of only one cluster of similar trajectories, based on manual analysis of the trajectories. In that case, cutting the dendrogram at a predetermined height will probably result in a relatively high number of clusters of similar size, as demonstrated in Figure 49.

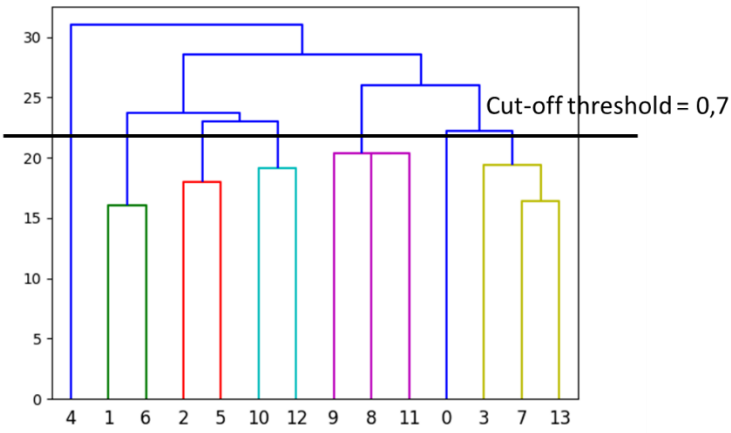



Figure 49: Shortcomings of threshold cut-off

***Elbow method***

The elbow method is based determining the knee or elbow of an error curve through visual evaluation of the error plot. This error-curve is constructed by dividing the data set in k cluster, for k ranging from 1 to n. This is done by cutting the dendrogram horizontally in a way the dendrogram is divided in k branches. For each value of k, the error is calculated by taking the squared sum of the distance between all objects in the data set to their cluster centroid. An example of such an error curve is shown in Figure 50.

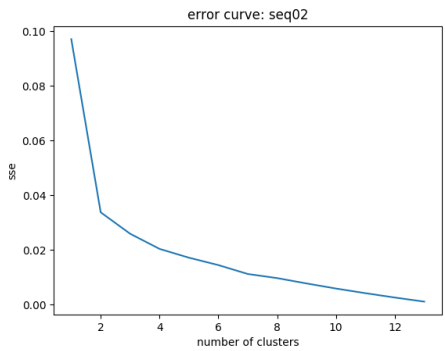



**Figure 50: elbow method error curve**

This error curve typically looks like an arm or leg. The point of maximal curvature of this curve is called the elbow or knee of the plot. This point represents the optimal number of clusters in the data set.

The idea behind this method is the following. When increasing the number of clusters, each trajectory in the data set will be closer to the centroid of its cluster. Past the elbow of the error-curve, the benefits of adding an extra cluster is significantly decreasing.

The elbow method, although less arbitrary than the threshold cut-off method, performs less well in cases where the data is not clearly clustered. In that case, the error curve is a rather smooth curve in which it is difficult to determine the optimal number of clusters. An example of the error-curve of such a data set is shown in Figure 51.

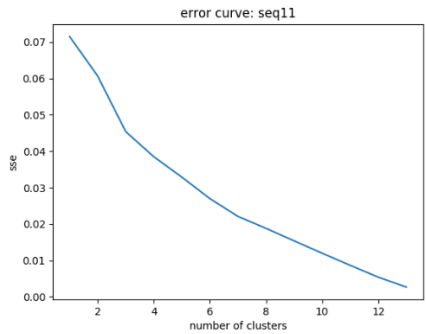



**Figure 51: elbow method error curve: flat similarity structure**

There exist a number of methods that aim to automate the evaluation of the error curve. The L-method [129] makes use of the property that both the parts left and right of the elbow are often approximately linear. Therefore, for every possible value of k, two lines are fitted through the error curve, one through the points on the left side of the presumed elbow and one through the points on the right side. These two lines intersect at the elbow of the curve. The pair of lines that most closely fits the curve, determines the optimal number of clusters in the data set. The best-fit is calculated using the least squared method. The L-method is further explained in Figure 52.

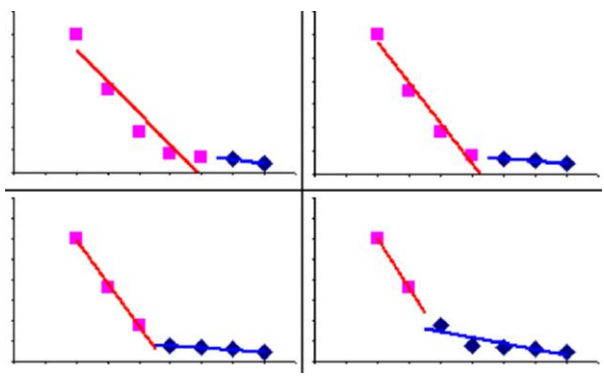


**Figure 52: L-method: finding knee of error curve**

***Gap statistic***

The GAP-statistic also makes use of the plot of an error measure. The basic idea is to standardize this error graph and compare it to its expectation under an appropriate null reference distribution of the data [130].

In this case, let $\{L_i\}$, I = 1,…,n be our data set consisting of n different trajectories. Then $d_{ii'}$ is the squared distance between trajectory i and i'. Suppose this data set is clustered into k clusters $C_1$, $C_2$, …, $C_k$, with $C_r$ denoting the indices of the trajectories in cluster r and $n_r = |C_r|$.

Now we can calculate the pooled within-cluster sum of squares around the cluster means as follows:

$$W_k = \sum_{r=1}^{k} \frac{1}{2\, n_r} D_r \tag{19}$$

where $D_r$ is the sum of all pairwise distances between all trajectories in cluster r:

$$D_r = \sum_{i,i' \in C_r} d_{ii'} \tag{20}$$

$W_k$ is evaluated for a different number of clusters k. The optimal number of clusters is determined by the value of k for which the normalized error curve $\log(W_k)$ falls the furthest below the expected error curve of the reference curve.

$$Gap_n(k) = E_n^*\{\log W_k\} - \log W_k \tag{21}$$

Where $E_n^*$ describes the expected value under a sample size n from the reference distribution. In this case, the reference distribution consists of trajectories of which the pairwise distance is uniformly distributed between the minimum and maximum pairwise distance between the trajectories in the data set.

To estimate $E_n^*\{\log W_k\}$, we calculate the average of B copies $\log W_k^*$, each of which is from a Monte Carlo sample $X_1^*, X_2^*, \ldots, X_n^*$ drawn from the reference distribution. As a result, we need to take into account the sampling distribution of the gap statistic. If sd(k) describes the standard deviation of the B Monte Carlo sample replicates $\log W_k^*$, then the simulation error $s_k$ in the estimation $E_n^*\{\log W_k\}$ can be calculated:

$$s_k = \sqrt{\left(1 + \frac{1}{B}\right) sd(k)} \tag{22}$$

Using this correction, we consider the smallest value of k for which $GAP(k) \geq GAP(k+1) - s_{k+1}$. This arbitrary 1-standard deviation rule has been used before in literature [131] and was experimentally validated by [130].

The outline of the GAP-statistic method is provided in Figure 53.

**GAP statistic**

```
Step 1
for k from 1 to K:
    calculate W_k
End for
Step 2
for b from 1 to B:
    generate reference data set
    calculate W*_kb
end for
calculate Gap(k) = (1/B) Σ_b log W*_kb − log W_k
Step 3
for k from 1 to K:
    calculate sd_k
    calculate s_k
end for
choose k̂ = min(k) for which Gap(k) ≥ Gap(k+1) − s_k+1
```

**Figure 53: GAP-statistic method outline**

***Tree Cut through permutation testing***

The three methods for determining the optimal dendrogram cuts described above, are all based on simple horizontal cuts. When the structure of the dendrogram is more complex, these methods might yield poor clustering results. To overcome this problem, Bruzesse [132] suggested to slice the dendrogram using permutation testing. The basic idea behind the method is that if all elements of two clusters are mixed together and split up randomly, the distance between the newly created clusters should not be significantly different from the distance between the original clusters. This idea was used to develop a more dynamic tree cutting method which is capable of making cuts at different heights in order to efficiently handle nested clusters in the data set.

The method starts from the root of the dendrogram where all trajectories in the data set are grouped into one single cluster. Then the method moves down the tree with a partial threshold until a new link joining two clusters is encountered. At each of these nodes, a permutation test is performed to investigate whether the two clusters that arise when cutting the branch at that node, must be accounted as a unique group (null hypothesis) or not (alternative hypothesis). In the case the null hypothesis cannot be rejected, the two clusters will form one single cluster in the final partitioning of the dataset and their sub-branches will not be further explored. If this null hypothesis is rejected, each of the two branches of the dendrogram will be further investigated in the procedure. The algorithm stops if there are no more branches left for which the null hypothesis can be rejected.

The principle of the permutation test can be explained as follows. Each branch under investigation is split up into two clusters at its root node: $C_L^i$ and $C_R^i$.

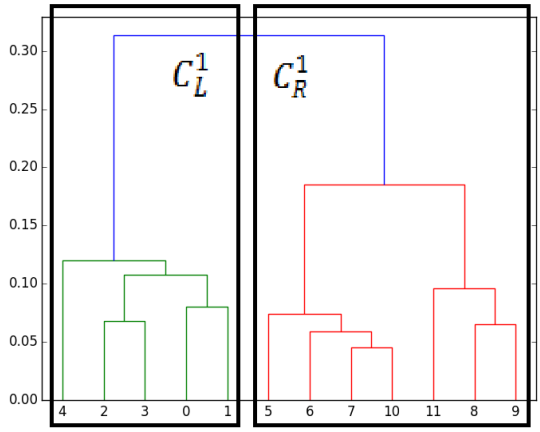



**Figure 54: permutation method: branch cutting**

For these two clusters, the inter-cluster distance is calculated by calculating the distance between the clusters centroids. The calculation of these cluster centroids is done using the DBA trajectory averaging method proposed by [103]. Subsequently, these two clusters are mixed up together and randomly split up, with the only constraint that the group cardinality stays the same. This procedure is performed m times, each time calculating the inter-cluster distance of the newly formed clusters $mC_L^i$ and $mC_R^i$.

The basic idea behind this method is that, if $C_L^i$ and $C_R^i$ are two separate clusters, the inter-cluster distance between $C_L^i$ and $C_R^i$ will likely be higher than the inter-cluster distance between $mC_L^i$ and $mC_R^i$. Thus, when the number m is large enough, a monte carlo p-value can be calculated as follows:

$$p = \frac{\#\left[intercl\ distance\left(mC_L^i, mC_R^i\right) \geq intercl\ distance(C_L^i, C_R^i)\right] + 1}{M + 1} \qquad (23)$$

The null hypothesis can then be rejected when this p-value is lower than a predetermined threshold value. The outline of this method is provided in Figure 55.

```
TreeCut(C)
1.  Initialization
2.  BranchesToVisit := [ C ] //start with complete dendrogram
3.  Clusters := []
4.  While BranchesToVisit is not empty
5.          C_L^i, C_R^i = cut(BranchesToVisit[1]) //split up the first branch in the list
6.          If C_L^i ≡ C_R^i
7.                  Add C_L^i ∪ C_R^i to Clusters
8.          Else
9.                  Add C_L^i and C_R^i to BranchesToVisit
10.                 Remove C_L^i ∪ C_R^i from BranchesToVisit
11.         End if
12. End While
```

**Figure 55: Dynamic tree cutting algorithm outline**

### *Tree cutting methods: summary*

In this section, four different dendrogram cutting or slicing methodologies have been discussed. Each of these methods has been implemented and tested on different data sets in order to find the best-fitting method for classifying human operator trajectories. The results of these experiments are described later on in this chapter. Table 6 provides an overview of these methods and summarizes the advantages and disadvantages of each algorithm.

**Table 6: overview of tree cutting methods**

| | unsupervised | parameter independent | nested clusters | calculation time |
|---|---|---|---|---|
| **treshold cut-off** | ✓ | × | × | ++ |
| **elbow method** | × | ✓ | × | + |
| **gap statistic** | ✓ | ✓ | × | - |
| **permutation testing** | ✓ | × | ✓ | -- |

## 3.2. Methodology for Human Operator Trajectory Clustering

In this section, we develop a suitable classification and outlier detection method for human operator trajectories through a series of experiments.

Agglomerative hierarchical clustering is used as the clustering algorithms because it is parameter-independent, capable of handling various similarity measures and does not require any prior knowledge about the number of clusters and outliers in the trajectory data set. In a first experiment, a combination of different linkage criteria and similarity measures is evaluated using an experimental data set. In a second series of experiments based on two different data sets, a suitable dendrogram cutting method is selected and validated.

### 3.2.1. Experiment 1: selecting linkage criterion and similarity measure

***Design of experiments***

The performance of the agglomerative clustering method can be influenced by both the applied (dis)similarity measure and the linkage criterion used to decide which clusters to merge. Also, these two factors might influence each other, meaning that a specific similarity measure yields good clustering results in combination with single linkage AHC, but leads to poor results when combined with the ward's linkage criterion. Therefore a full factorial design was set-up, based on the experimental data set presented in the previous chapter.

This data set consists of 11 different scenarios. Each of these scenarios were manually analyzed. This way every trajectory was either assigned to a cluster or labeled as an anomalous event. The manual analysis is considered to be the correct solution of the clustering problem and serves as a basis for validating the developed clustering methodology. An overview of the different scenarios in the dataset is provided in Table 7.

**Table 7: Experimental data set: overview**

| | clusters | outliers |
|---|---|---|
| **scenario 1** | 1 | 0 |
| **scenario 2** | 1 | 0 |
| **scenario 3** | 1 | 2 |
| **scenario 4** | 2 | 0 |
| **scenario 5** | 1 | 1 |
| **scenario 6** | 1 | 1 |
| **scenario 7** | 3 | 2 |
| **scenario 8** | 1 | 1 |
| **scenario 9** | 3 | 1 |
| **scenario 10** | 2 | 0 |
| **scenario 11** | 1 | 0 |

For each of these scenarios AHC was applied using five different (dis)similarity measures in combination with the four linkage criteria mentioned above. Based on our literature review, the similarity measures that were selected are: One-way distance (OWD), Fréchet distance, Dynamic Time Warping distance (DTW), Hausdorff distance and longest common subsequence (LCSS).

For each scenario, this analysis leads to 20 different clustering dendrograms. Subsequently, horizontal cuts were made in these dendrograms to divide the data set into clusters and outliers. Cuts are made until the sum of outliers and clusters is equal to the sum of outliers and clusters in the ground truth obtained through the manual analysis. For example, 2 horizontal cuts were made in the dendrogram for scenario 3. This way the trajectories in this scenario are divided into three groups or single element clusters. In this experiment we use knowledge about the real data structure and number of clusters, which would normally not be available. However, this experiment aims to determine which combination of AHC method and distance measure provides the best similarity structure. By making these cuts manually, we are able to assess whether the resulting dendrogram represents the actual similarity structure of the data set using the evaluation methods described in the following sections.

By performing this procedure for every possible combination of linkage criterion and similarity measure, 20 different classifications of the same set of trajectories were performed. These classifications are then compared to the ground truth which is determined through the manual analysis.

***Evaluation of clustering methods***

Four different evaluation criteria were used to assess the quality of the classification obtained through a specific combination of linkage criterion and similarity measure.

- **Rand Index Score (RI):** RI is measure of the accuracy of the clustering method. For every pair of trajectories in a data set, the classification method has to make the decision whether these two trajectories are actually similar or not. If the classification method assumes similarity, it will return a positive result, otherwise it will output a negative result. Compared to the ground truth, this result can either be false or true. Therefore, each decision made by the classification algorithm can be considered to be true positive, true negative, false positive or false negative, as explained in Table 8.

**Table 8: classification evaluation**

| | | ground thruth | |
|---|---|---|---|
| | | same class | different class |
| classification output | same cluster | True Positive (TP) | False Positive (FP) |
| | different cluster | False Negative (FN) | True Negative (TN) |

The rand index score calculates the ratio of true decisions to the total number of decisions taken. In this case, the total number of decisions is equal to the total number of trajectory pairs in the data set. The rand index score can be calculated as follows:

$$RI = \frac{TP + TN}{TP + TN + FP + FN} \tag{24}$$

- **Precision:** The precision measure is used to assess the chance that a predicted positive result is actually true. High precision means that trajectories that are grouped in the same cluster by the classification method, will be similar in reality. It also means that there is only a small chance that the proposed method will assign anomalies to clusters containing regular trajectories. The precision score is however no indication whether the method tends to unnecessarily split up larger clusters into smaller groupings of regular trajectories.

  The precision of a classification method is calculated by dividing the number of correct positive results by the total number of positives returned by the method.

$$P = \frac{TP}{TP + FP} \tag{25}$$

- **Recall:** Recall or "probability of detection" is a measure of the sensitivity of a classification method. In our case, it calculates what percentage of similar trajectory pairs actually results in a positive output from the classifier. A high recall percentage means that there is only a small chance that the classification method will split up clusters which only consist of similar trajectories.

  To calculate the recall score, the true positive results are divided by the total number of positives in the data set.

$$R = \frac{TP}{TP + FN} \tag{26}$$

- **F-measure:** The F-measure is the harmonic mean of the recall and precision scores. It is calculated as follows:

$$F = (1 + \beta^2)\ \frac{P \cdot R}{\beta^2 \cdot P + R} \quad \text{where } \beta > 0 \qquad (27)$$

  Depending in the β value, the F-measure attaches more importance to the recall (β >1) or precision (β <1). In this section we use the $F_1$-measure (β=1).

***Experimental results***

The detailed results of this analysis are added in appendix 1. Figure 56 and Figure 58 below visualize these results. Figure 56 shows the average performance of the different similarity measures for all linkage criteria. Figure 58 shows the performance of the different linkage criteria, averaged over the different similarity measures.

On first sight, the Fréchet distance and DTW seem to perform consistently better than the other similarity measures. When looking at the different linkage criteria, single linkage seems to be outperforming the other criteria. However, a quick look at the resulting dendrograms shows that the single linkage criterion results in rather flat clustering results, decreasing the ability of this method to make a clear discrimination between different clusters (chaining effect). This is visualized for a number of scenarios in Figure 57.

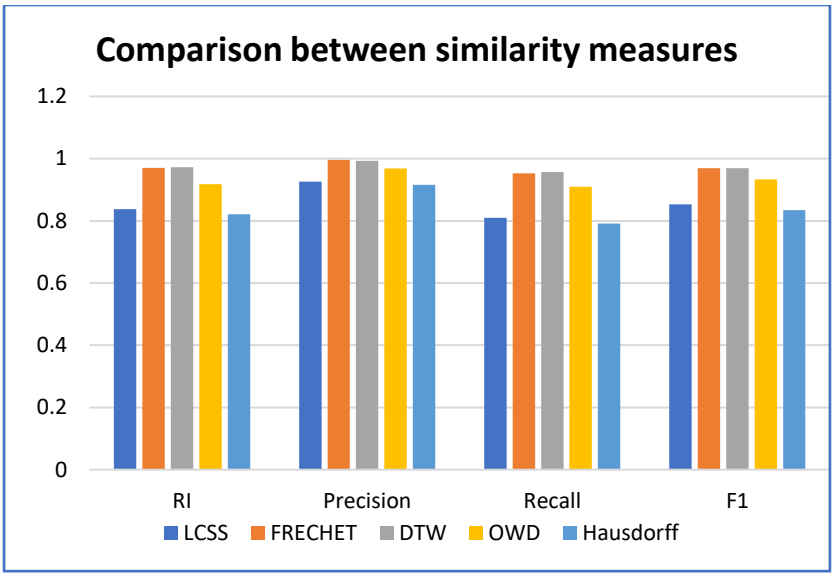



**Figure 56: evaluation of similarity measures**

WARD's linkage criterion appears to yield the worst overall results. But it's capability to detect outliers, indicated by the precision measure, seems to be at the same level of its counterparts. Drawing conclusions just based on these results is very difficult, therefore a series of statistical tests is performed to check similarities and differences between different similarity measures and linkage criteria. To test the statistical significance of the observed differences obtained through different clustering methods, a Friedman test was applied on the results of this experiment. The Friedman test is a non-parametric procedure that is often used in a hypothesis testing situation involving a design with two or more samples and is often used as an alternative for ANOVA [133]. It can be used as a multiple comparison test that aims to detect significance in the observed difference in the behavior of two or more algorithms. The reason to choose the Friedman test above a regular ANOVA test, lies in the distribution assumption of ANOVA. In ANOVA, the dataset is supposed to be normally distributed. This is clearly not the case for this experiment.

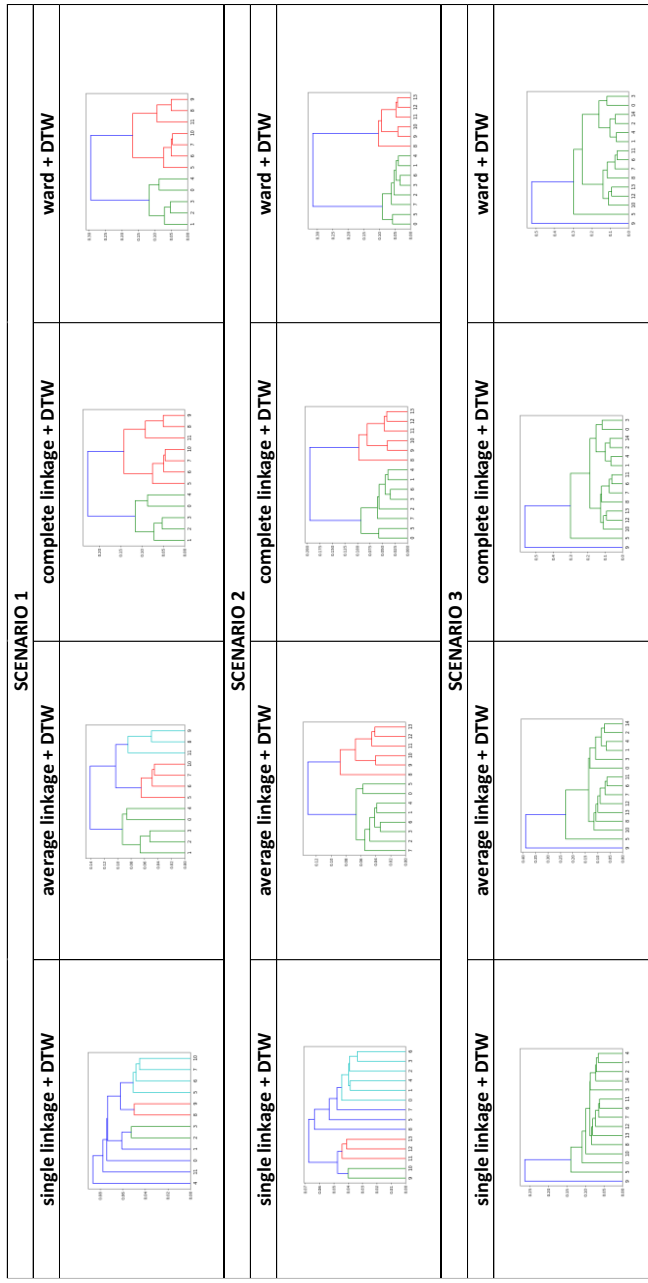



**Figure 57: linkage criteria: discrimination capability**

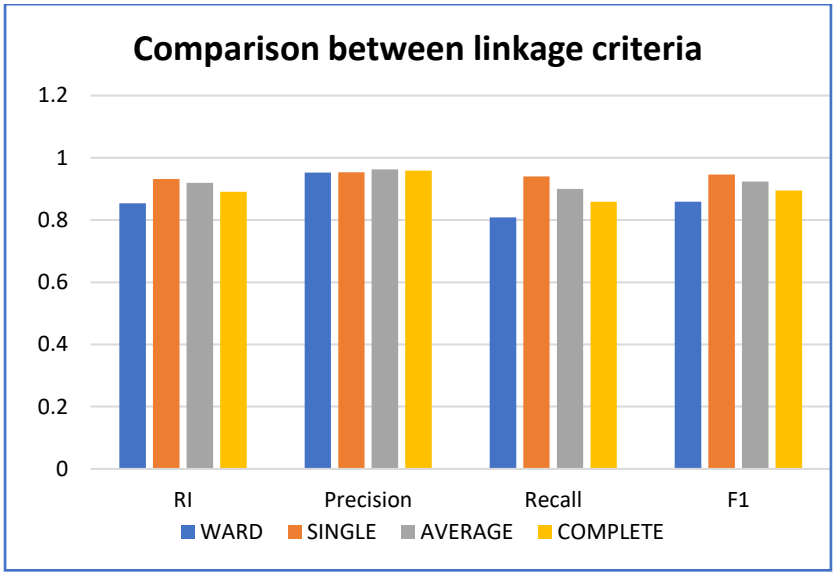



**Figure 58: evaluation linkage criteria**

The null hypothesis tested using Friedman's test is $H_0$: $\theta_1 = \theta_2 = ... = \theta_k$ : the median of population i represents the median of population j for 1≤ i ≤ k and 1 ≤ j ≤ k. In other words: the results obtained through one algorithm in the experiment are also representative for the results obtained through any other algorithm in this experiment. The alternative hypothesis is $H_1$ = not $H_0$. In the case this is true, there exists a significant difference between the results, which means that one or more algorithms perform remarkably better or worse than the others.

This Friedman test was performed four times, each time using a different evaluation criterion as the algorithms' result. For every scenario in the dataset, the results of each algorithm were ranked, starting at 1 for the best result. Under the null hypothesis, based on the assumption that all proposed algorithms are equivalent and therefore there rankings are as well, the Friedman test statistic can be calculated:

$$\chi_F^2 = \frac{12\,N}{k\,(\,k+1)}\left[\sum_j R_j - \frac{k\,(k+1)^2}{4}\right] \qquad (28)$$

Where $R_j = \sum_i r_i^j$, N equals the number of cases or scenarios considered for every algorithm and k represents the number of algorithms tested in the experiment. Once N ≥ 10 and k ≥ 5, the critical values of this test statistic coincide with the ones established in the $\chi^2$ distribution.

The detailed results for this test are summarized in Table 9 below.

**Table 9: Friedman test results 20 methods**

| FRIEDMAN TEST RESULTS | | | | |
|---|---|---|---|---|
| | **RI** | **Precision** | **Recall** | **F1** |
| **N** | 11 | 11 | 11 | 11 |
| **df** | 19 | 19 | 19 | 19 |
| **test statistic Qf** | 83.1 | 82.1 | 84.7 | 84.7 |
| **significance level** | 5.4E-10 | 8.0E-10 | 6.2E-10 | 2.9E-10 |

The critical value for this experiment, using a significance level α=0.05, is 30.144 [134]. For every evaluation criterion, the calculated test statistic exceeds this critical value. The null hypothesis can thus be rejected, meaning that there is a significant difference in classification performance between the different tested methods.

Since the Fréchet and DTW distances seem to perform better than the other similarity measures, the same test was performed only withholding the results obtained by using these two measures. This new Friedman test only has 7 degrees of freedom which means that the critical value in this case is 14,067. The results of this analysis are summarized in the following Table 10.

Table 10: Friedman test results condensed test

| FRIEDMAN TEST RESULTS | | | | |
|---|---|---|---|---|
| | **RI** | **Precision** | **Recall** | **F1** |
| **N** | 11 | 11 | 11 | 11 |
| **df** | 7 | 7 | 7 | 7 |
| **test statistic Qf** | 11.7 | 9.8 | 12.2 | 11.7 |
| **significance level** | 1.1E-01 | 2.0E-01 | 9.6E-02 | 1.1E-01 |

According to these results there appears to be significant difference in performance between the Fréchet distance and DTW. Also, there is no noticeable difference between the results obtained through one of the four linkage criteria. Therefore, the performance of both DTW and Fréchet distance as well as all linkage criterions in combination with different tree cutting methods, will be investigated.

***Experiment 1: conclusion***

In this experiment, five different similarity measures and four linkage criteria were are evaluated. Based on the results obtained through a full factorial design, where each of these measures was tested in combination with each of the linkage criteria on 11 different sets of trajectory data, it can be concluded that the DTW and Fréchet distance measures perform better than shape-based measures such as Hausdorff and one-way distance. Therefore we will only use DTW and Fréchet distance in further experiments.

The aim of this experiment was to select suitable similarity measures as well as the best linkage criterion for agglomerative clustering. However, it is hard to draw any conclusions on which linkage criterion to use. On average, single linkage seems to outperform the other criteria. A closer look into the results on the other hand, shows that the combination of complete linkage and dynamic time warping is the only method that performs flawlessly in this experiment. Also, the dendrograms generated through single linkage clustering show that the discrimination ability of this method is rather low. Ward's linkage

criterion turns out to be the least promising method, but performs as good as any other method when it comes to its ability to detect outlying trajectories.

It is clear that similarity measures and linkage criteria cannot be treated separately. This gives us good reasons to assume that also the best way to make the final decision on the dendrogram cutting height, is depending on the similarity measure and linkage criterion it is combined with. Therefore a new experiment was designed in order to find the most suitable combination of these three factors.

### 3.2.2. Experiment 2: Determining the dendrogram cut-off method

***Design of experiments***

To determine which is the best dendrogram cut-off methodology, a new series of experiments was designed on the same experimental data set as used in experiment 1. In this case, the number of cluster in the sets of trajectories, is considered to be unknown. From the previous experiment we learned that both DTW and Fréchet distance seem to be suitable similarity measures. In this experiment all combinations of these two measures, the four linkage criteria and four different dendrogram cut-off methods are investigated.

For evaluating the results of this experiment, we focus on the two most important performance measures: random index score and precision. These two measures respectively quantify the accuracy of the clustering algorithm and its capability to detect outliers, which is the main goal of our classification framework.

***Experimental results***

The results of this experiment are summarized in Table 11. Out of the 16 methods tested in this experiment, there are 5 methods that yield an accuracy and precision of over 90%. These methods are indicated in gray.

Table 11: Evaluation tree cutting methods

| | DTW | | FRECHET | |
|---|---|---|---|---|
| | **Treshold cut-off (0.7)** | | | |
| **Linkage** | **RI** | **Precision** | **RI** | **Precision** |
| **single** | 0.69 | 0.83 | 0.66 | 0.85 |
| **average** | 0.75 | 0.92 | 0.71 | 0.96 |
| **complete** | 0.79 | 0.92 | 0.82 | 0.98 |
| **ward** | 0.74 | 0.92 | 0.79 | 0.98 |
| | **Elbow Method** | | | |
| **single** | 0.78 | 0.95 | 0.74 | 0.98 |
| **average** | 0.87 | 0.97 | 0.76 | 0.98 |
| **complete** | 0.83 | 0.97 | 0.82 | 0.98 |
| **ward** | 0.78 | 0.94 | 0.80 | 0.98 |
| | **GAP Statistic** | | | |
| **single** | 0.82 | 0.97 | 0.90 | 0.96 |
| **average** | 0.79 | 0.92 | 0.83 | 0.96 |
| **complete** | 0.79 | 0.92 | 0.81 | 0.96 |
| **ward** | 0.70 | 0.92 | 0.78 | 0.96 |
| | **Dynamic tree cut: permutation testing** | | | |
| **single** | 0.91 | 0.98 | 0.89 | 0.95 |
| **average** | 0.81 | 1.00 | 0.84 | 0.96 |
| **complete** | 0.80 | 1.00 | 0.92 | 1.00 |
| **ward** | 0.90 | 0.93 | 0.94 | 0.99 |

Overall, the dynamic tree cutting algorithm based on permutation testing outperforms the other dendrogram slicing methods. In contrast to the results of the first experiment, ward's linkage criterion performs rather well with both similarity measures. However, making a well-funded choice between similarity measures and linkage criteria is still impossible based on the experiments with these small data sets.

***Validation of the results***

The second experiment shows that there are a number of classification methods that yield promising results. While some methods seem to have a very good outlier detection capability, their overall accuracy seems to be inferior to that of some other methods.

It is clear that a trade-off between precision and accuracy needs to made. Therefore, as a rule of thumb, we only consider those methods that achieve a score of 90% or higher on both performance indicators as relevant candidates. These candidate methods are: DTW-single linkage, DTW-Ward linkage, Frechet-complete linkage and Frechet-Ward linkage

The experiments described in previous sections were performed on rather small experimental data sets. To decide on the most suitable classification framework for human operator trajectories, we want to test candidate methods on larger data sets as well. Therefore, the performance of the selected methods was validated using the Omnilab data set, which contains over 200 trajectories distributed over 15 different clusters. The results of this test are plotted in Figure 59.

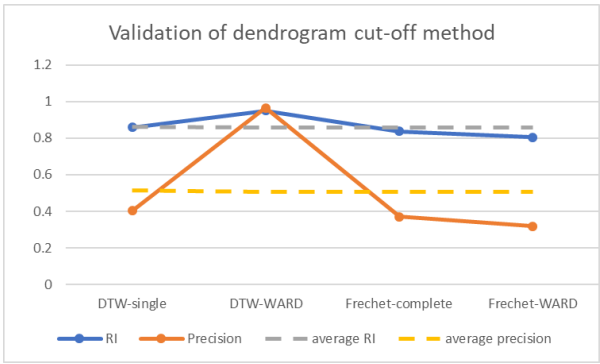



**Figure 59: Validation classification framework**

***conclusion***

The results of this experiment show that, on average, the dynamic tree cutting procedure using permutation testing clearly results in superior clustering results.

Based on the results shown in Table 11 and Figure 59, this dendrogram cutting method works most consistently in combination with dynamic time warping and using Ward's linkage criterion. Throughout the different experiments, this combination of linkage criterion and similarity measure almost never seemed to be the absolute best

method. However it almost always results in satisfying clustering results and it is less depending on the case under investigation.

The reason for this is that single and complete linkage consider single-point measurements to determine the distance between clusters, while ward's linkage aims to minimize variance within the cluster and takes all elements in that cluster into account. This strongly reduces the chance for chaining effects or the risk that one single outlying trajectory prevents the merge of two nearby clusters. The major drawback of the Fréchet distance is the fact that the similarity calculation is solely based on the closeness of the values independently of the local trends. In other words, in this case the Fréchet distance only takes into account the locations visited by the operator, but does not consider the time the operator spent at a specific location. DTW does make this alignment, resulting in better clustering results in some cases.

### 3.2.3. Parametrization and robustness analysis of the classification framework

The focus of this chapter is the development a classification framework that is able to detect outliers in a data set of human operator trajectories. In the ideal case, this framework works unsupervised, is independent of any parameters and is capable of handling complex data sets. The similarity measures and linkage criteria under investigation all comply with these requirements. However, the dynamic tree cutting method proposed relies on two parameters, the p-value and the number of monte carlo simulations M.

To check how these parameters influence the clustering results, the framework has been tested using different values for these parameters. In permutation testing, typical confidence levels used are 95% and 97%, resulting in p-value thresholds of 0.05 and 0.03.

It is clear that the number of monte carlo replications M influences the calculation time. Therefore, a trade-off between M and the calculation time needs to be made. This was done by running the classification method for a number of different values of the

parameter M. Each of these runs was replicated 10 times and average values of the random index score, precision and calculation times are exhibited in Figure 60 and Figure 61 for p equals 0.03 and 0.05 respectively.

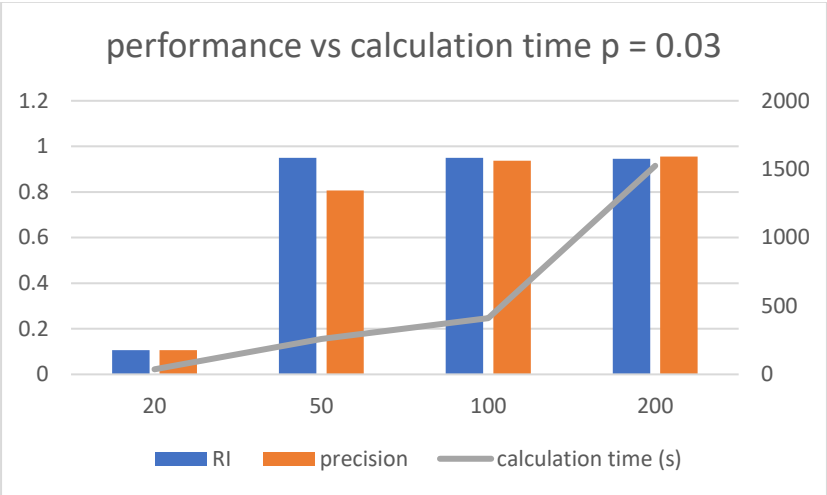



**Figure 60: clustering performance vs calculation time p=0.03**

The calculation time increases exponentially with the number of monte carlo replications. The selected confidence level has only a limited influence on the precision and accuracy level. As expected, the outlier detection capability increases as the p-value increases, but the effects are only marginal. The very low performance observed for p-value threshold = 0.03 and M = 20 can be easily explained by the way the p-value is calculated. Even if the monte carlo simulation doesn't return one single positive result, the calculated p-value equals:

$$p = \frac{1}{20+1} = 0.047 > 0.03 \qquad (29)$$

In this case, the trajectories in this data set will always be considered to belong to the same cluster. The calculation also shows that increasing M decreases the influence of a coincidental positive result of the monte carlo simulations. These graphs indicate that, in this case, a setting the parameter M at 100 provides the desired classification results in a reasonable time span. Further increasing this value doesn't really contribute to the performance of the

classification method while it increases the calculation time exponentially.

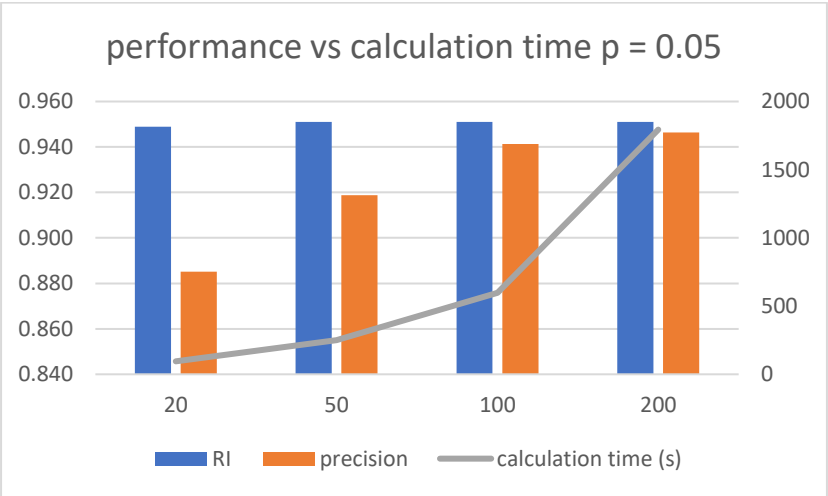



**Figure 61: clustering performance vs calculation time p=0.05**

The parameter setting does not only influence the average performance, but also the variation on the clustering results. In order to make sure that classification framework consistently provides acceptable clustering results, this variation needs to be minimal.

In Figure 62 the average performance of the classification method is plotted for a p threshold value of 0.05 together with its variation over the different calculation runs for M values ranging from 20 to 200. These results are also exhibited in Table 12.

**Table 12: clustering variability**

| M | RI | Precision | sd RI | sd Precision |
|---|---|---|---|---|
| **20** | 0.949 | 0.885 | 0.003 | 0.070 |
| **50** | 0.951 | 0.919 | 0.002 | 0.057 |
| **100** | 0.951 | 0.941 | 0.002 | 0.003 |
| **200** | 0.951 | 0.946 | 0.001 | 0.010 |

Again we can conclude that a M-value of 100 provides consistent clustering results. Note that the selection of a parameter setting of M

will be depending on the number of elements in a data set, the number of clusters and the variation in cluster cardinality within that data set. The optimal parameter setting will thus need to be determined case by case.


**Figure 62: Clustering variability**

## 3.3. Implementation of the Clustering Framework

The framework described in this chapter is implemented in Python 2.7, using Eclipse Neon®. The agglomerative clustering method used is part of the open source SciPy python library. Since ward's linkage criterion is not implemented in this library, also the python Fastcluster library was used.

The remaining functions of the framework are all programmed specifically for this research. Code snippets of the implementation of all similarity measures are provided in appendix 2.

## 3.4. Output of the Clustering Framework

The output of the clustering method divides a data set of human operators in different clusters. Objects that are classified in single-item clusters are considered outliers. For each of the clusters, average trajectories are calculated. Examples of the output for a number of clusters in the Omnilab data set are visualized in Figure 63 and Figure 64.

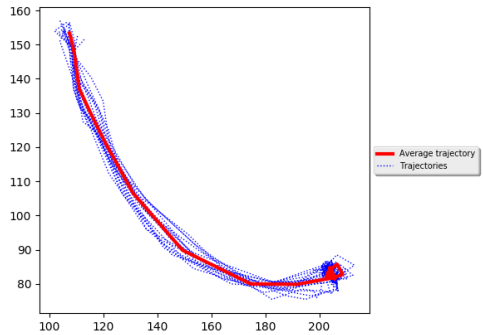



**Figure 63: Clustering output example 1**

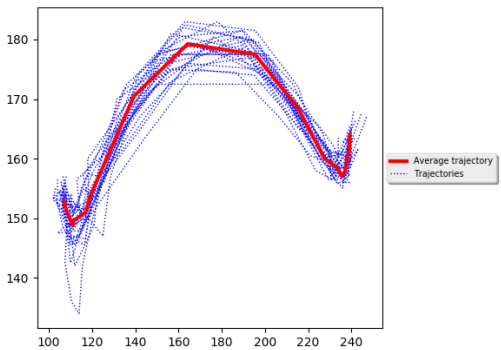



**Figure 64: Clustering output example 2**

The average trajectories of a cluster serves as a model for the specific work cycle represented by the trajectories in this cluster. These models will later on be used as a basis for the real-time monitoring and outlier detection framework, discussed in Chapter 4. After the classification procedure, average work cycles and best practices are calculated for every cluster/pattern. These best practice work cycles, together with the event list, can serve as automatically generated work instructions for the operator and serve as a benchmark for real-time operator performance measurement.

One of the main assets of the clustering method is its ability to detect outlying patterns, which usually indicate problems or irregularities in

the operators' work flow. After clustering, outlying trajectories can be linked to their originating video images. This way, the analysis time of a stream of video images can be significantly decreased by pointing directly to the interesting fragments in the video footage.

## 3.5. Conclusions

In this chapter, a framework for human operator trajectory clustering and outlier detection is presented. The framework relies on agglomerative hierarchical clustering methods to structure the dataset based on the trajectory similarity. To asses this similarity, trajectories are aligned through a dynamic time warping procedure and similarity score is calculated. The final partitioning of trajectories into clusters is done using a statistical tree cutting algorithm that is based on permutation testing.

This framework was developed through a series of experiments on experimental data sets. First a literature study was performed to identify existing similarity measures, clustering methods and tree cutting algorithms which could potentially be applied on human operator trajectories. These methods were implemented and assessed on their accuracy and capability to find anomalous events in the dataset. These experiments showed that the proposed framework outperforms the other methods in both performance and consistency.

The framework presented in this chapter is capable of capable of classifying human operator trajectories which are deduced from video images, with an accuracy and outlier detection rate of over 90%. Detected outliers can afterwards be linked to the original video images and fed back to the operator or analyst to find the cause of the problems. This provides the method engineer with an approach to efficiently analyze long video streams with minimal effort. Also, the resulting models of the normal trajectories can serve as a basis for real-time monitoring purposes, as will be explained in the next chapter.

# Chapter 4 Real-time anomaly detection and performance measurement

In this chapter, we investigate how we can use the operator monitoring and trajectory classification framework to support assembly operators and production managers in real-time. The first section focuses on the development of a real-time trajectory matching methodology which uses the results of the classification framework presented in chapter 3. Real-time task recognition enables online operator support through the use of an operational dashboard. In the second section of this chapter we evaluate existing manufacturing dashboard concepts and present a new dashboard (OAWSAD) concept to better support continuous improvement on the shop-floor level. The outcome of this chapter is summarized in discussed in section 3.

## 4.1. Real- Time Classification and Outlier Detection

The classification method described in the previous chapter, results in clusters of normal trajectories and some outliers. For each of the clusters, the average trajectory is calculated. These average trajectories provide models of normal trajectories which can be observed frequently. These models can serve as a template to recognize tasks and detect issues or outliers in real-time.

A naïve approach to do this, would be to compare every incoming trajectory to all calculated models. This comparison could be based on the DTW distance and the variance in the cluster could be used to set threshold values to determine whether the observed trajectory matches one of the models. However, there are two major issues to this approach: (1) DTW can only be calculated for the full trajectories and (2) DTW has a O(n*m) calculation time complexity. Performing these calculations in real-time for a number of different models is

impossible. Therefore, a real-time matching procedure, based on a lower-bound calculation method for DTW, was developed.

### 4.1.1. Keogh Lower bound calculation

The real-time outlier detection and classification method relies on the DTW lower bound calculation as proposed by Keogh and Ratanamahatana [135]. The basic idea behind the method is to compare the incoming sequence to a subsequence of the generated models based on a low complexity lower bound calculation. These subsequences of the previously calculated models have the same length as the incoming sequence. The incoming trajectory sequence only represents a fraction of the full work cycle. One can rightfully question whether matching such a partial sequence to the model of a complete work cycle would actually provide meaningful results.

To calculate this lower bound of the DTW distance, a bounding envelope is generated for every trajectory [136]. Let $M(a_1, a_2, \ldots, a_m)$ be a trajectory model of length m, then we calculate the bounding envelope Env(M) by generating two different time series Up(M) and Low(M) as follows:

$$Up(M) = \max\left(M_j \middle| j \in [\max(1, i-b), \min(m, i+b)]\right) \quad (30)$$

$$Low(M) = \min\left(M_j \middle| j \in [\max(1, i-b), \min(m, i+b)]\right) \quad (31)$$

In other words, $Up_i(M)$ and $Low_i(M)$ are respectively the maximum and minimum values of M in the interval [i-b, i+b], where b is a user-defined parameter and taking into account the border effects.

The squared Keogh lower bound distance between the incoming trajectory $S_n$ of length n and an equally sized subsequence $M_n$ of the model $M(a_1, a_2, \ldots, a_m)$ with n<=m, is calculated as follows:

$$LB_{Keogh}(Env(M), S)^2 = \sum_{i=1}^{n} \begin{cases} \left(S_i - Up_i(M)\right)^2 & if\ S_i > Up_i(M) \\ 0 & if\ Low_i(M) \leq S_i \leq Up_i(M) \\ \left(S_i - Low_i(M)\right)^2 & if\ Low_i(M) > S_i \end{cases} \quad (32)$$

It can be proven that $LB_{keogh}$ distance is a lower bound for the DTW distance in the case of one-dimensional time series. The trajectories under consideration in this research, are however 2-dimensional. Therefore, the lower bounds for both the x- and y-component of the trajectory are calculated separately. Rath and Manmatha [137] proved that in this case:

$$\begin{aligned} LB_{Keogh}(Env(M_x), S_x)^2 \\ + \ LB_{Keogh}\left(Env\left(M_y\right), S_y\right)^2 \\ \leq \ DTW(M_x, S_x)^2 \\ + \ DTW\left(M_y, S_y\right)^2 = DTW(M, S)^2 \end{aligned} \tag{33}$$

The calculation of the Keogh lower bound is further explained in Figure 65 and Figure 66 for the x and y component of a sequence and model in the Omnilab dataset.

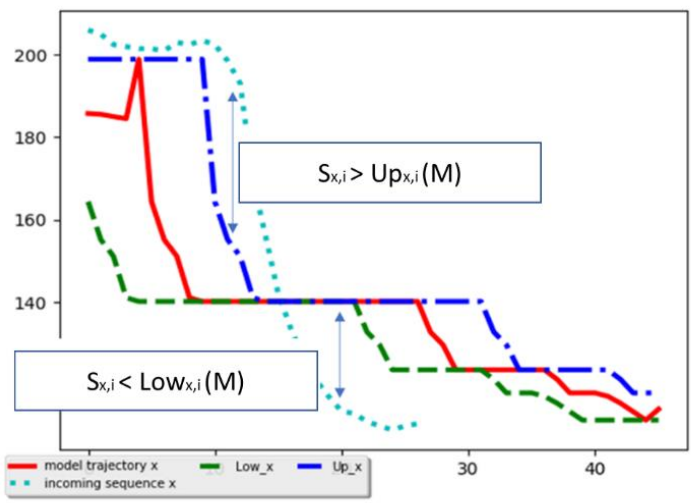



**Figure 65: bounding envelope x**

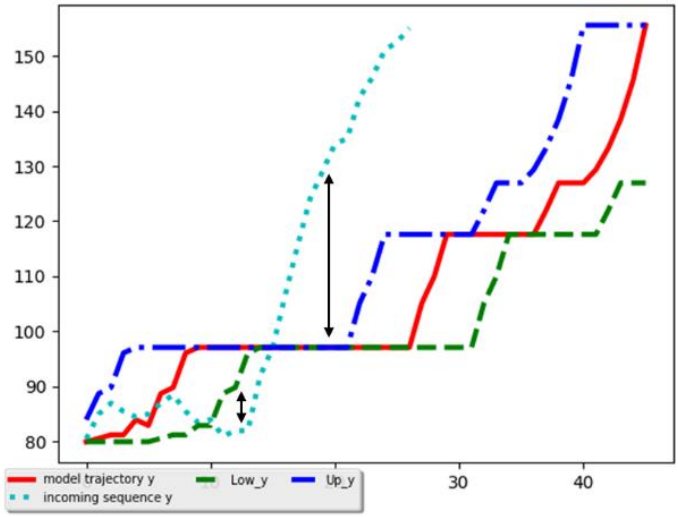



Figure 66: bounding envelope y

### 4.1.2. Real-time trajectory matching

Based on the $LB_{keogh}$ concept, a real-time trajectory matching method was developed. When the real-time location system starts returning a new trajectory, the $LB_{keogh}$ distances between the incoming trajectory and an equally sized subsequence of all model trajectories are calculated. The model resulting in the lowest $LB_{keogh}$ distance is considered to be the provisional best match (best_so_far) and for this subsequence, the real DTW distance is calculated. Subsequently, the LB distance values are compared to this best_so_far DTW distance and all models for which LBkeogh > best_so_far are removed from the list of candidate matching models, under the assumption that those models are unlikely to provide a good match for the incoming sequence. This procedure is repeated as the incoming sequence is growing and as there are still multiple models in the list of candidate matches. The outline of this method is provided in Figure 67.

**Real_Time_Trajectory_Matching(M: [($x_1$, $y_1$), …, ($x_m$, $y_m$)], S: [($x_1$, $y_1$), …, ($x_n$, $y_n$), …])**

```
Initialization
Best_so_far ← inf.
Incoming_traj ← []
Start
For new data point:
        LBs := [Keogh_LB for model in traj_models]
        Best_so_far = DTW(model with best LB)
        For model in traj_models:
                If LB>Best_so_far:
                        Remove from traj_models
                End if
        End for
        Return best model
End for
```

**Figure 67: outline real-time trajectory matching algorithm**

As shown in Figure 67, the algorithm uses the Keogh LB to guess the best matching model. This way, only one DTW calculation needs to be performed per new incoming data point. The algorithm was then further sped up by implementing a warping window for the DTW calculation. Such a warping window restricts the range of the time deformation that can be used to match the sequences.

To detect outliers, the average distance between the average trajectory of the best matching model and all the trajectories used to build up that model (*avg_dist*), is calculated together with the standard deviation $\sigma_d$ on those distances. Once the incoming trajectory is fully completed, the DTW distance between the new trajectory and the average trajectory of the best matching model is compared to *avg_dist*. If DTW(incoming, avg_traj) > avg_dist + z. $\sigma_d$ the incoming trajectory is considered to be an outlier.

Figure 68 and Figure 69 show what the output of the method looks like for a normal and outlying trajectory respectively through a number of screenshots. The incoming sequence is shown in blue, the best matching model in the set of normal trajectories is drawn in red. When a match is found, this is indicated by a green background, when

the best matching model is not similar to the incoming trajectory, the red background color indicates that a problem is detected.

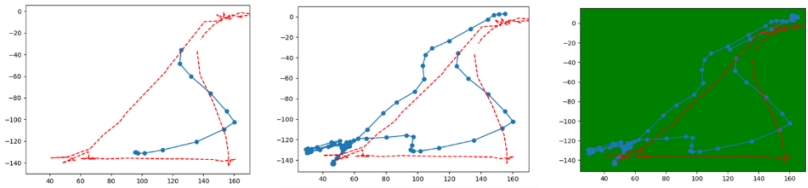


Figure 68: Real-time trajectory matching algorithm - matching trajectory

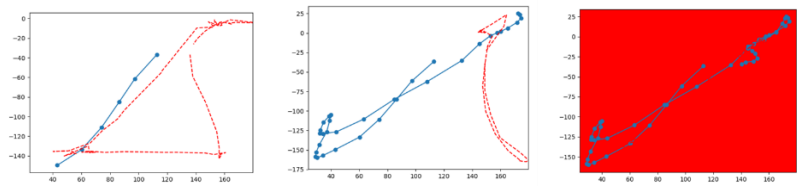


Figure 69: Real-time trajectory matching algorithm - outlying trajectory

### 4.1.3. Validation of the real-time trajectory matching procedure

To validate the real-time outlier detection and classification procedure described in the previous paragraph, an artificial data set was created. This data set uses models from different scenarios in the experimental data set. Scenario 1 is used to create the first model of a normal trajectory, the second model was build using the trajectories of one of the two variants in scenario 10. The created models together with the individual trajectories used to generate these models, are visualized in Figure 70 and Figure 71

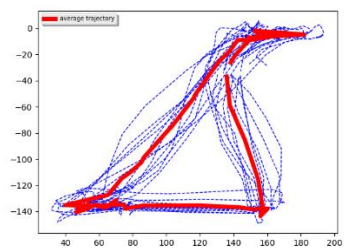



Figure 70: model based on sequence 1

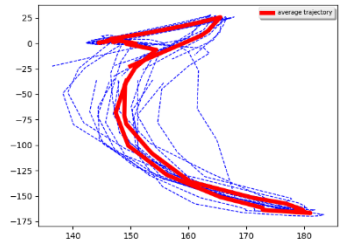



**Figure 71: model based on sequence 10**

After training the models, the trajectories were presented to the framework and compared to these models. To mimic the framerate of the camera system, one location point per 0.2 seconds was added to the incoming trajectory. This corresponds to a vision system which works at a framerate of 5 frames per seconds, which is higher than what the current vision system can provide. On the other hand, we showed in chapter 3 that we could safely sample the trajectories to a frame rate of 2 samples per second, without losing classification performance. For each new location point added to the trajectory, the calculation time was determined to check whether the method is capable of performing all calculations in real-time.

***Simulation run 1:***

In a first simulation run, trajectories returned by the simulated vision system are the same trajectories used to train the models. This set of trajectories was extended with two outlying sequences taken from scenario 9 to test the framework's outlier detection capability. This

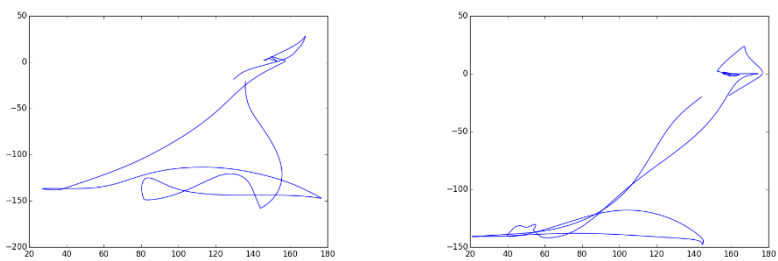



**Figure 72: outlying trajectories**

data set thus contains 2 outliers and a set of normal trajectories of two different patterns.

For this simulation run, the method filters out the two outlying sequences and matches the other trajectories to the right model. Since the incoming trajectories in this test are the exact same trajectories used to generate the models of normal patterns, we expect that the method yields a high classification accuracy. Using test sequences that are part of the training set, leads to overfitting. Therefore this simulation runs provides no information on the validity of the method, but helps to test whether the implementation is done correctly and provides information on the expected calculation times. On average, the calculation time per frame for this simulation run is 0.0613 seconds, with a maximum calculation time of 0.07314 seconds.

***Simulation run 2:***

In sequence 2 of the experimental data set, the exact same tasks as sequence 1 were performed by a different operator without any issues. Therefore, the trajectories in sequence 2 should be comparable to the ones in sequence 1. Using the same models used in the first simulation run, a new simulation run was set-up. The trajectories returned by the vision system are the trajectories of sequence 2. Since the test trajectories are not used to generate the models, the chance of detecting classification mistakes grows.

This is however not the case, again all trajectories are matched to the right model in an average time of 0.072 seconds, which is still acceptable considering a required framerate of 2 frames per second.

***Simulation run 3***

To test how well the method behaves when there are multiple models, we extracted a training and test object set from the Omnilab data set. To create the models, we used trajectories of the 7 most appearing patterns in the data set. Per pattern, at least 5 trajectories were used to create the model. Next, we selected the leftover trajectories describing those same patterns, to create a set of 50 test objects.

Because there are more models in the data set, one would expect that the average calculation time would increase. However this is not the case: in this test run the average calculation time amounts to 0.0678 seconds. The reason for this is the fact that the calculation time is still heavily determined by the DTW calculation for the best fitting model. Since the models in this data set contain less location points, the time required for this calculation decreases.

On the downside, applying the proposed method on this data set only yields an accuracy 94.3% percent. In this simulation run, no false negatives (no matching pattern was found when it does exist) were detected. The 5.7% mistakes detected are trajectories that are matched to the wrong model (false positive). This can be explained by the fact that some of the models in this data set share common sub-trajectories. Sometimes this results in a slightly higher similarity of the incoming trajectory to a sub-trajectory of the wrong model. This occasionally leads to the preliminary elimination of the actual best matching model.

## 4.2. Operational Assembly Work Station Analysis Dashboard

### 4.2.1. Dashboards in manufacturing environments

Target-oriented and real-time information provisioning across all hierarchy levels is a critical success factor for a manufacturing company to attain agile and efficient manufacturing processes [138]. Operational performance dashboards are a tool that is often used by manufacturing companies to effectuate this information transfer. Dashboards are a means to measure performance and initiate continuous improvement initiatives. By visualizing KPI's and setting targets, dashboards help operators and production managers to focus on achieving these targets and identifying improvement potential [139]. Just as an automobile dashboard controls and directs the behavior and decisions of a car driver, operational manufacturing dashboards are a critical tool to manage manufacturing operations.

Since the beginning of this millennium, we have seen an upgoing trend in the use of dashboards across manufacturing companies [140]. The use of these dashboards is perfectly complementary to the PDCA approach often used in manufacturing companies. The Deming cycle, also known as the PDCA (Plan-Do-Check-Act) cycle, is an iterative four-step lean management tool used to facilitate continuous improvement of processes and products [141]. The four stages of the method are the following [142]:

- **Plan:** Identify issues and potential root causes, propose alternative methods to overcome issues.
- **Do:** Implement solutions, measure the results.
- **Check:** Analyze the results, measure effectiveness and decided whether implemented solution overcomes the issue.
- **Act:** Make adaptations to the solution if necessary, standardize improved method.

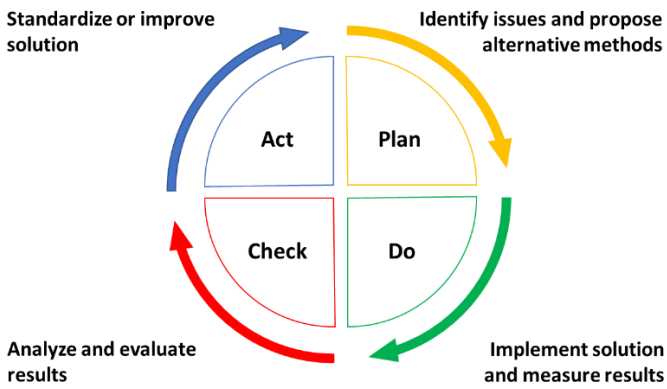



Figure 73: PDCA methodology for continuous improvement

By providing accurate and up-to-date process information, dashboards have become a valuable support tool for all manufacturing stakeholders (operators, production managers, sales, customers, etc.) in their pursuit of operational excellence. Dashboards are extremely helpful to identify gaps between targets and performance, identify root causes and analyze the effects of adaptations made to the process.

Their exist many different types of dashboards in the context of manufacturing. Depending on the hierarchy level they are used on, these dashboards incorporate different performance metrics. According to Gröger [143], manufacturing dashboard concepts in industry can be categorized in three groups, each acting on a different strategical level: (1) Business activity Monitoring dashboards, (2) Manufacturing Control Panels and (3) Operational Process dashboards.

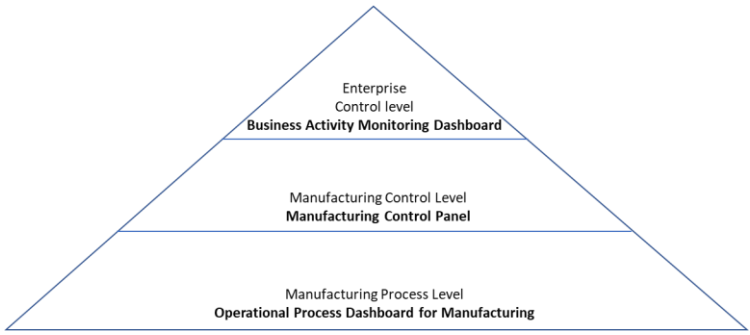



**Figure 74: Classification of manufacturing dashboard concepts [143]**

Business Activity Monitoring (BAM) dashboards are typically used on the enterprise control level and mainly focus on monitoring critical business processes and generating alerts when action is needed. Examples of indicators that are typically monitored in a BAM dashboard are sales, customer satisfaction, income and expenses, etc. According to Yusof [140], these dashboards are used for strategic decision-making on the long term.

Manufacturing control panels are typically used by production managers on the operations control level. They are typically used to support detailed scheduling, capacity and resource planning, process monitoring (f.e. quality, maintenance).

Operational process dashboards are designed to support decisions on the very detailed process level. They are typically shown at the work station and contain information that is relevant for the operators in the work station. Many examples of operational dashboards can be

found in literature [144-146]. However, most of them only provide high level process information such as machine utilization and product quality. Also commercial providers of business intelligence software solutions offer operational dashboard solutions. SAP has a standard Production Operator Dashboard in their product portfolio. But this dashboard is mainly focused on data acquisition and providing digitized work instructions.

To overcome these issues, Gröger et al. [143] developed a generalized operational process dashboard for manufacturing (OPDM) to support the workers on the shop floor. The dashboard mainly aims to extend the services of existing dashboards with process knowledge such as video-based work instructions and a platform to improve communication on both shop-floor and enterprise level. However, the performance component of their dashboard still contains rather high level information on which operators only have limited influence, as shown in Figure 75.

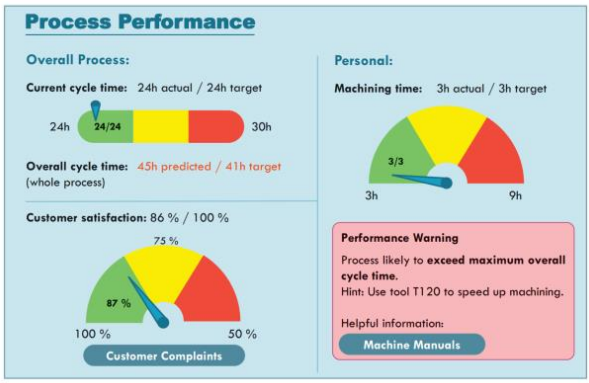



**Figure 75: Process performance dashboard presented by Gröger [143]**

The effectiveness of manufacturing dashboards is influenced by both the information it contains and the way this information is represented. In a survey of dashboard design methodologies and dashboard implementations, Yusof and Othman [140] discovered a number of common features that are beneficial for the effectiveness

of the dashboard. First-of-all, dashboards should contain real-time information. Secondly this information should be visualized using clear graphs and charts that are easy to interpret and lastly, the users should be able to retrieve this information on different aggregation levels (task level, job level, day, shift, …) to support root cause analysis.

In literature, most research on these dashboards address the design aspects and IT issues [147-150]. Tokola et al. presented a survey in which they identified relevant KPI's for manufacturing companies and the time frames and hierarchy level where these KPI's should be used [138]. However, the resulting Operational Dashboard for Workers merely provides insights in the state of the production system and lacks drivers for improvement (Figure 76).


**Figure 76: Operational dashboard for workers as presented by Tokola [138]**

Through a review of existing commercially available dashboards [151-154], we identified which information is commonly used in operational manufacturing dashboards currently used in manufacturing environments:

- Count: number of products produced in a certain time interval
- Target: Expected number of products produced over a certain time interval. By showing count and target numbers, a high level evaluation of the performance can be made.

- Scrap: Proportion of bad or scrap pieces produced in a certain time interval.
- Rework: Number of reworked parts in a certain time interval.
- Overall Equipment Effectiveness (OEE): measures how effective production equipment is being used.
- Down time: Indicates what proportion of the available time equipment or a work station is unavailable due to failures, breakdowns, lack of material, etc…
- Labor Efficiency Variance: Measures the difference between the standard cost of actual direct labor hours utilized and the expected standard hours necessary to achieve the output realized.
- Injuries: Indicates the time period without work-related injuries caused by accidents. Showing this number creates awareness among workers about health and safety risks in their work environment.

Although their relevance cannot be denied, these KPI's act on a more managerial level and provoke almost no actions for improvement in the work station itself. Both operators and production managers have little or no impact on these indicators. For this reason, the effect of these dashboards on the process level.

### 4.2.2. Dashboard information

To overcome the issues with existing dashboards described above, a new operational assembly work station analysis dashboard (OAWASD) concept was developed. As mentioned earlier, method study often relies on the use of graphs, diagrams and charts to identify issues and their root causes. These charts are rather simple tools that provide transparency and better insights into various process parameters. Together with some of the traditional performance indicators, a selection of relevant charts is integrated in the OAWSAD to drive continuous improvement on the work station and shop floor level.

The dashboard is based on information generated by the real-time trajectory classification and monitoring framework presented earlier.

By linking the OAWSAD to real-time trajectory information, charts and performance indicators can automatically be generated and updated with minimal effort. Because it provides real-time information, all users can evaluate the effects of implemented changes almost immediately, speeding up the PDCA cycle significantly.

The dashboard can be used by both assembly operators as production managers and team leaders. The remainder of this section provides an overview of these charts and performance indicators. As a conclusion of this section, the OAWSAD is presented.

**TAKT:** takt time is the maximum amount of time in which a product needs to be finished in order to meet the customer demand. Takt means pulse in German, by using the takt concept, a natural rhythm is created across all processes. In assembly lines, work stations are balanced against the takt time. That means that every work station in the line should be able to finish every work cycle within takt requirements. If not, the assembly line needs to be stopped and valuable production time goes to waste. Visualizing TAKT and comparing it to the actual performance of the work station, helps to detect capacity problems and process synchronization issues.

**YAMAZUMI CHART:** Yamazumi charts are often used as a visual management tool in Lean implementations [155] . A Yamazumi chart is a stacked bar chart that visualizes the total cycle time of work stations or operators. Yamazumi charts can be generated for different aggregation levels. By comparing the Yamazumi charts for all work stations in the line, the charts support production managers to identify and solve line balancing issues. In work stations with multiple operators, charts per individual worker can trigger the redistribution of tasks to achieve a balanced work load per operator.

In our dashboard solution, we developed a yamazumi chart to compare the actual performance of the operator to the targeted process time. This is especially interesting for assembly work stations with very long cycle times, where takt monitoring is often difficult. The developed chart allows operators to evaluate in the meantime if they will be able to finish their work cycle within TAKT by showing the expected time buffer. Once this buffer drops below zero, the column

will turn red to indicate that corrective actions need to be taken. The chart can be used in real-time by the operator. When for example the operator encountered a problem with a specific task, he/she can quickly evaluate whether he will still finish in time or needs to ask help from the team leader or so-called butterfly workers. These are highly skilled operators which are typically used in assembly lines to provide some flexible work force capacity. An example of such a Yamazumi chart is provided in Figure 77.

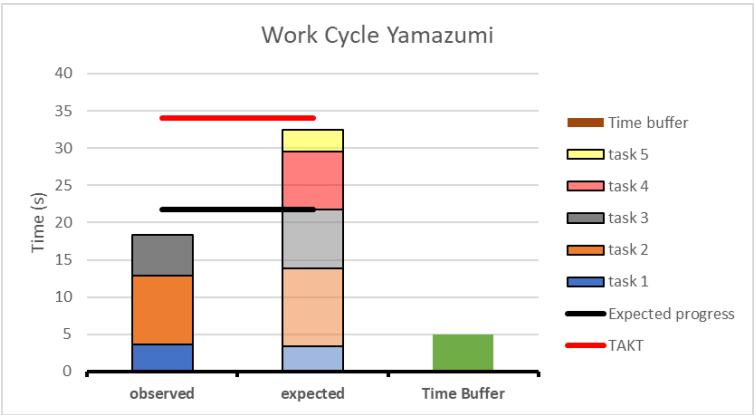



**Figure 77: Work Cycle Progress Yamazumi Chart**

The use of real-time trajectory data offers makes it possible to keep these Yamazumi charts up-to-date with real-time observations and updated time standards. The use of the real-time trajectory classification method previously described offers some extra advantages over existing commercial monitoring solutions. Since the monitoring system recognizes the task done by the operator, there is no need for a fixed assembly sequence (in contrast to f.e. the Arkite HIM® system). This way, the operator gets the freedom to change the assembly sequence whenever he thinks it is beneficial, for example when parts for the next task are missing or when the operator believes he can improve the process by using a different assembly sequence.

Besides the Yamazumi chart shown in Figure 77, other Yamazumi chart options on different aggregation levels are added to the dashboard. For example, a chart to evaluate the work distribution

between two workers in the same work station and a line balancing chart are available for the user.

**STATISTICAL PROCESS CONTROL (SPC) CHARTS:** SPC charts are one of the main tools used in the six sigma quality management strategy. Control charts allow the user to analyze how the process performance changes over time. It is mainly applied to quality measurement, but there are also examples in literature where SPC charts were used for process control in assembly environments [156]. The main goal of control charts is to determine whether a process is in control or not. A process is in control of the observed variability is random and a natural part of the process. For out-of-control processes, this variability is no longer random and patterns can be discovered. These patterns are caused by specific external sources and require further investigation.

SPC charts are constructed by plotting the evolution of the observed variable (in this case, task time) over time, together with the average value. Six extra lines are added to the chart, the Upper and Lower control limit (UCL and LCL), typically 3 times the standard deviation above and under the average value respectively. All these control values are calculated based on a test run or historical data. Also the $\bar{x} \pm 2\sigma$ and $\bar{x} \pm \sigma$ limits are added to create six control zones, as shown in Figure 78.

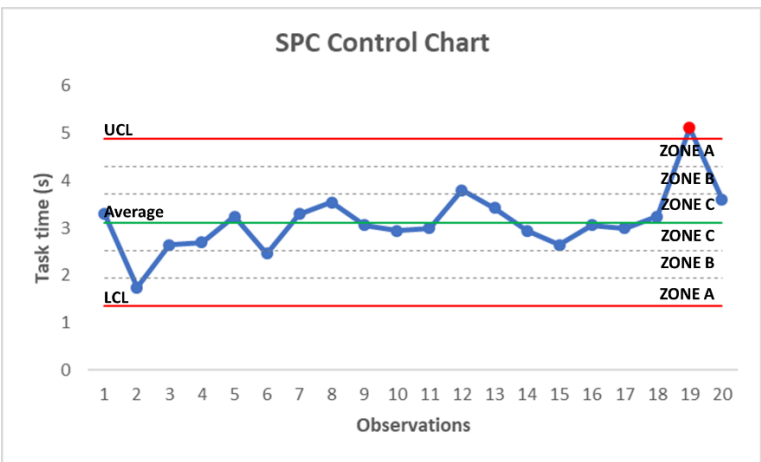



**Figure 78: SPC control limits**

To determine whether a process is in control or not, there are 8 control rules that indicate when special causes of variation are present. These rules are explained in Table 13.

**Table 13: Control chart rules**

| Rule | Rule Name | Pattern |
|---|---|---|
| **1** | Beyond limits | One or more points beyond the control limits |
| **2** | Zone A | 2 out of 3 consecutive points in Zone A or beyond |
| **3** | Zone B | 4 out of 5 consecutive points in Zone B or beyond |
| **4** | Zone C | 7 or more consecutive points on one side of the average |
| **5** | Trend | 7 consecutive points trending up or down |
| **6** | Mixture | 8 consecutive points with no points in Zone C |
| **7** | Stratification | 15 consecutive points in Zone C |
| **8** | Over-Control | 14 consecutive points alternating up and down |

In the proposed dashboard, SPC charts can be generated on the level of individual tasks or complete work cycles. When one of the above patterns is recognized, the dashboard will trigger an alert to initiate corrective actions. In assembly work stations, SPC charts can indicate many different issues. Some examples are:

- Order sequencing issues: when multiple variants of a product with a high workload for a specific task are scheduled consecutively, this can lead to balancing issues in the line.
- High variability for a specific task can indicate a lack of operator training for that task.
- A trend of increasing task time can be an indicator for fatigue.
- A Zone A or Zone B violation can be an indication the operator is not well trained for a specific task.

These examples are mainly scheduling and organizational issues, therefore these SPC charts are more likely to be used by the production manager.

Besides pointing to issues, these charts can also show positive effects. When linked to the real-time trajectory data for example, when a method is improved, a Zone A or Zone B pattern below the average

might be discovered. This pattern can serve as a trigger to adapt the time standards based on the latest observations.

**SPAGHETTI CHARTS / WIRE DIAGRAMS:** A spaghetti diagram is a quick and easy way to track distances of people or sometimes parts on the shop floor. In typical assembly environments, especially larger work stations, operators spend a significant amount of their time walking from one point to another. Since this time is considered to be non-value-adding, it should be reduced as much as possible. Spaghetti charts have been used industry to reduce operator movement with great success, especially in repetitive environments such as assembly work stations [157, 158]. They are typically used by production managers to evaluate the layout of the work station, but they can also trigger the operator to adapt his behavior (f.e. taking a shorter path or changing the task sequence to reduce 'unloaded' walking).

Spaghetti charts typically start from a layout and are manually drawn while observing the operator in the work station [159]. The spaghetti charts in our dashboard start from the annotated layout described in chapter 2. Because spaghetti diagrams can sometimes be cluttered and unclear, different colors are used to discern separate tasks. Based on the calculation of the POI's and the number of moves between those POI's, a relationship graph is added to the spaghetti diagram. The weight of the vertices in this graph, indicates the frequency of certain moves between POI's. These graphs are often used in method study as a basis for graph-based layout optimization methods. The use of real-time operator trajectories makes it possible to automatically generate and update the chart. This way both the observation time and the feedback loop to evaluate process or layout changes are significantly reduced.

**TRAVEL DISTANCE:** Excessive movement is one of the main causes of inefficiency in typical assembly work stations. Operators constantly move around to pick parts, collect tools or gather information. Besides productivity losses, unnecessary movement induces fatigue and increases the risk of injuries. Therefore, reducing the travelled distance in the work station has a direct positive effect on the performance of the operator.

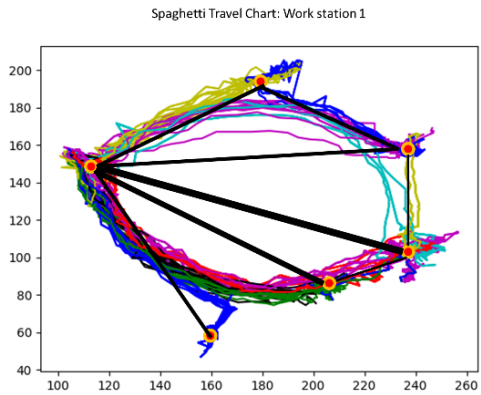



**Figure 79: example of a spaghetti chart**

Spaghetti charts make unnecessary movement visible and induce layout and process changes. By showing the distance covered by an operator during his/her shift and comparing it to the expected value, the effects of layout changes can immediately be evaluated. Therefore, the travel distance as a performance measure, is perfectly complementary to the spaghetti diagrams described earlier.

The travelled distance of the operator is calculated by taking the sum of the distances between all consecutive locations in the operator's trajectory.

$$travel = \sum_{i=1}^{n-1} \sqrt{(x_i - x_{i+1})^2 + (y_i - y_{i+1})^2} \qquad (34)$$

Where n equals the length of the trajectory. To avoid inaccuracies in this calculation due to noise in the raw trajectory data, distance calculations are done on the smoothed data, but before sampling. Low sampling rates come with the risk of underestimating the travelled distance, as shown in Figure 80.

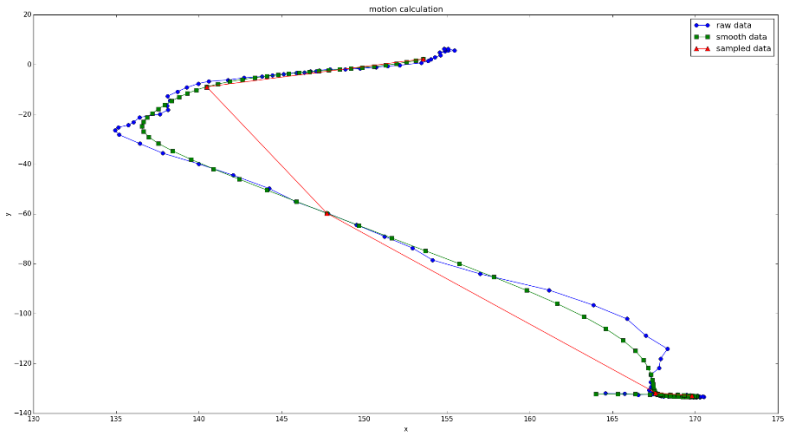



**Figure 80: effect of sampling on calculation accuracy**

**BALANCE DELAY:** Balance delay is a measure of the inefficiency in the assembly line, caused by idle time due to imperfect allocation of tasks among work stations [160]. The balance delay is an indicator of spare capacity in the production line. Through continuous improvement, task times should naturally decrease leading to an increasing balance delay. Increased balance delay figures can be a trigger to rearrange the production line (f.e. remove one work station). Balance delay for an assembly line is calculated as follows:

$$BD = \frac{n * TAKT - T_{nc}}{n * TAKT} \tag{35}$$

In this case n is the number of work stations in the line and $T_{nc}$ is the total measured task time for all work stations.

**SMOOTHNESS INDEX (SI):** Where the balance delay indicates spare capacity in the process, the smoothness index is an indication of how this capacity is distributed over the different work stations [160]. In most assembly lines, smooth balances with equal time buffers for every work station are pursued to guarantee robustness. The smoothness index calculated as follows:

$$SI = \sqrt{\sum_{i=1}^{n} (TAKT - T_{ci})^2} \quad (36)$$

$T_{ci}$ represents the observed work content for work station i. It is clear that the SI is an absolute measurement which depends on the number of work stations. Therefore it is difficult to compare SI figures between different assembly lines, however the SI can be useful for the production manager to evaluate newly implemented balancing schemes.

**VALUE ADDED RATIO (VAR):** An important concept in lean manufacturing is the difference between value added and non-value adding activities. The value added ratio indicates the proportion of time actually spend on adding value to the final product. Value added ratios are typically rather low, the VAR in work stations of a world class manufacturing company such as Toyota, rarely exceeds 20% [161]. The value added ratio is presented in the dashboard using a speedometer-like graph. It allows users to rapidly gain insights in the improvement potential present in the work station.

**RUN-AT-RATE (RAR):** When a work station is not capable of finishing its work cycle within TAKT time, it creates disturbance in the whole assembly line. In some situations, the line needs to be stopped with clear productivity losses as a result. In some cases, the operator is allowed to continue his tasks across the borders of his own work station, however this should be avoided as much as possible, because the extra time spent should be compensated on the next product to stay on schedule.

Since work cycles that exceed TAKT lead to direct productivity losses, it is an imported factor to monitor. The RAR factor is calculated as follows:

$$run - at - rate = \frac{\#cycles\_within\_takt}{\#cycles} \quad (37)$$

A low RAR score can have different causes: it can be due to poor line balancing or it can be caused by excessive variability of a specific work

element. The RAR factor can be visualized for a specific time interval (typically one shift), but in the dashboard it is also possible to show the RAR factor for one specific product variant to support root cause analysis. The choice to provide this information on the work cycle level and not on the individual task level was made because the inherent variability on task level would yield distorted results. In every work cycle there are tasks that take some more time than expected, but usually this time is compensated for by taking into account a time buffer while balancing the line or the operator can catch up this time during other tasks.

**OUTLIER RATIO:** Is a quality measure which indicates the proportion of work cycles that follow a normal pattern. It captures human errors such as picking mistakes that may arise due to the lack of standardized work instructions or poor operator training. These mistakes would not necessarily be detected in the quality measures that are typically used, because those measures mainly focus on the quality of the finished products. However, they lead to inefficiencies that negatively impact that performance of the work station. The outlier ratio is calculated based on the number of outliers detected by the classification framework.

$$outlier\ ratio = \frac{\#outliers}{\#cycles} \tag{38}$$

**WORK INSTRUCTIONS:** Besides monitoring, one of the main functions of an operational dashboard is to inform its users. Work instructions are the main source for assembly operators. Based on the event list described in chapter 2, we can automatically generate high level work instructions and link them to the video recordings of the best practice for that specific task. The real-time trajectory analysis framework makes it possible to assess different assembly sequences used by different operators and make suggestions on what is the best sequence to use. Together with textual and video instructions, the dashboard also visualizes the preferred path to follow for a specific task.

### 4.2.3. Operational Assembly Work Station Analysis Dashboard

All this information is integrated in an Operational Assembly Work Station Analysis Dashboard (OAWSAD) concept, which can be used by both assembly work station operators and production managers. Unlike most existing manufacturing dashboards, the proposed OAWSAD aims to provide its users with relevant process information and charts to drive continuous improvement on the shop-floor level. The dashboard was designed following the design rules proposed by Yusof and Othman [140].

Different users require different information on various aggregation levels. All this information is accessible through the dashboard. However, in order to keep it clear, we do not want to overload the user with information that is irrelevant to him/her. Therefore, the dashboard uses drop-down menus to allow the user to customize the dashboard to his/her needs.

The real-time trajectory analysis and classification methods presented earlier, enable us to keep the dashboard updated with real-time process information. All charts in the dashboard can be automatically generated and updated, which speeds up the feedback loop to evaluate improvement actions implemented by the operator or production manager.

An example of the OAWSAD is presented in Figure 81. The dashboard was built in excel and uses a standard input file generated by the Python framework. All information in the dashboard is color coded to allow the user to assess in a glance whether a work station is performing as it should. The threshold values used for the color coding, can be easily adapted to fit the needs of the user.

Figure 82 shows a second example for a simulated assembly line. The figure shows how the drop-down menus can be used to customize the dashboard and how this functionality can be used for root cause analysis. In this example, The RAR (A), Smoothness Index (B) and line balancing Yamazumi (C) indicate a clear balancing problem in Work Station 2. Looking at the Yamazumi chart (D) for this work cycle, one task appears to take significantly longer than expected. When

retrieving the SPC chart (E) for this specific task, a clear shift can be noticed for the last 5 iterations. This shift could for instance be due to the fact that a different operator who lacks the appropriate training, is manning the work station after a shift change.

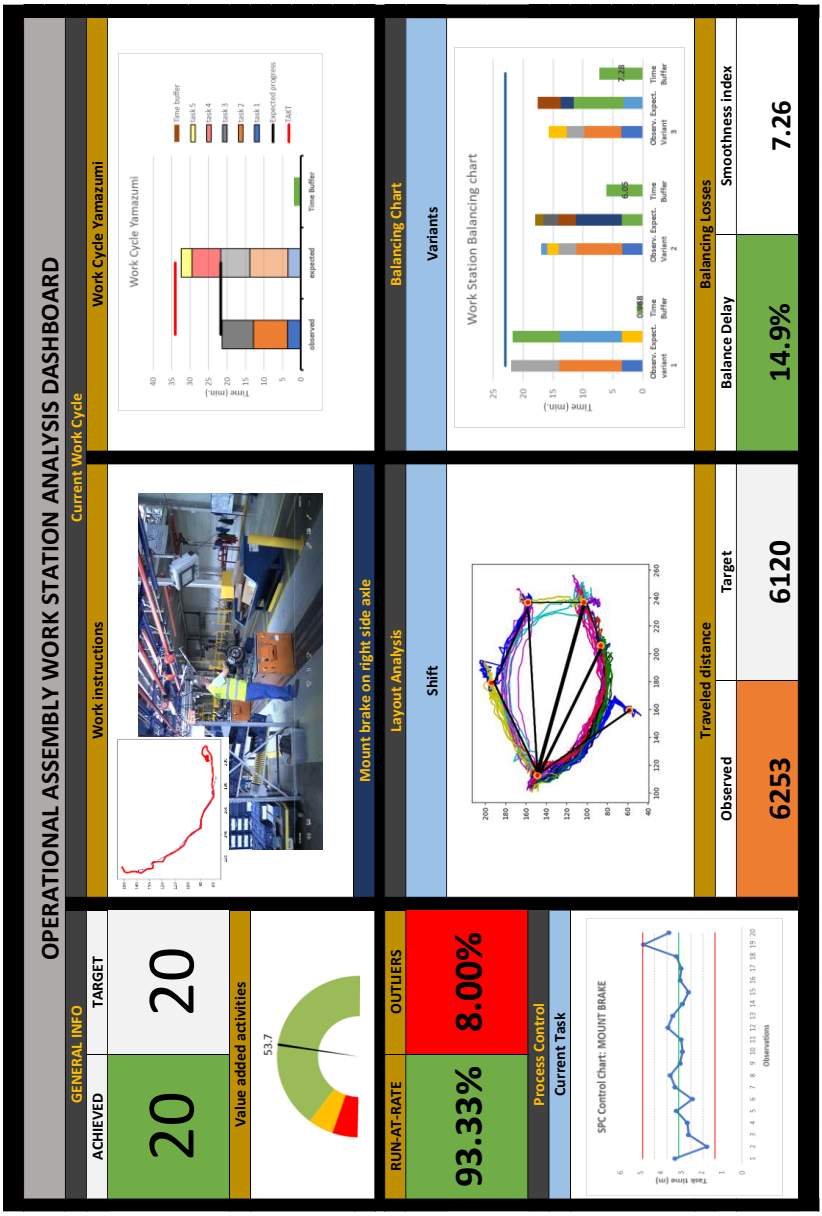



Figure 81: Example of the OAWSAD

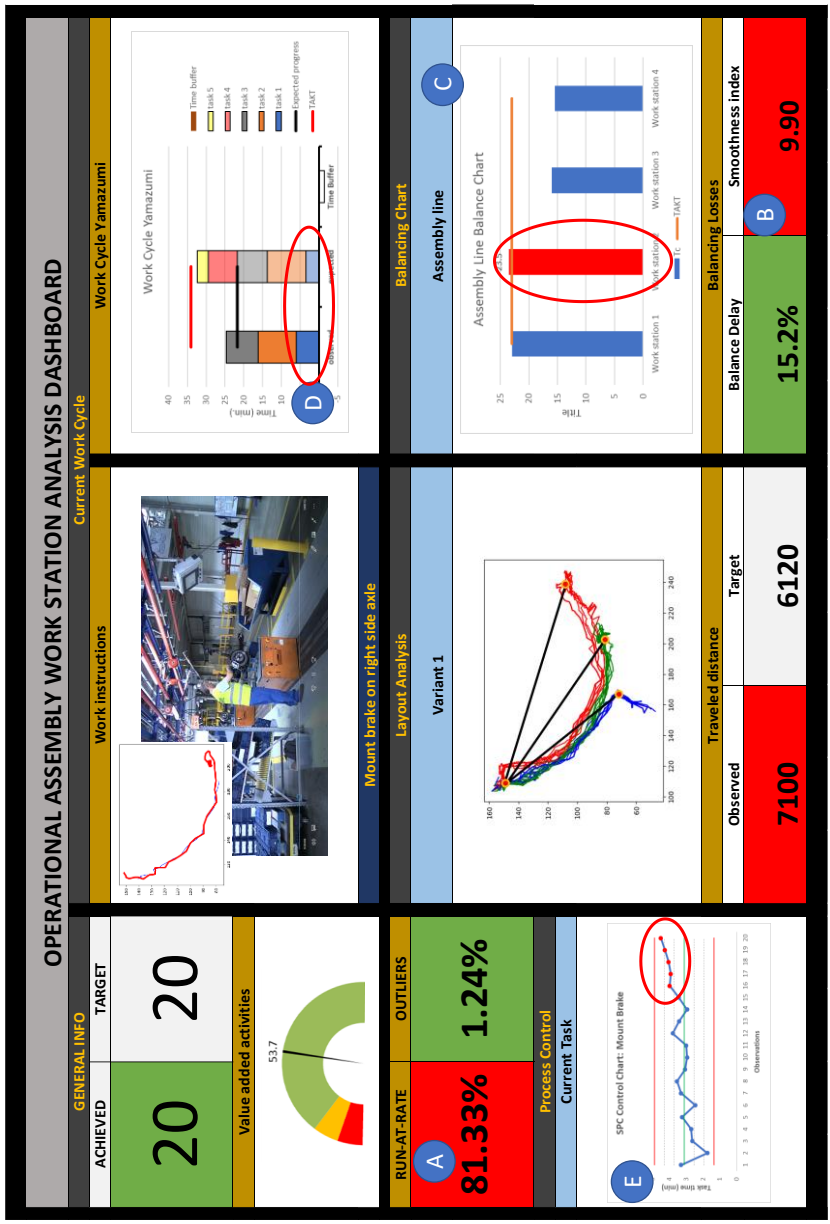



**Figure 82: OAWSAD balancing problem**

The experimental cases used to develop the real-time trajectory analysis and classification framework do not grasp the complexity of a real life production environment. Therefore we were not able to

investigate the effect of implementing such a dashboard in real assembly line work stations.

## 4.3. CONLUSIONS AND DISCUSSION

In the first section of this chapter, a real-time classification and outlier detection method was presented. The method starts from the models generated by the classification framework presented in Chapter 3. The method makes use of the Keogh lower bound concept and is capable of accurately matching incoming trajectories to predefined models in and detect issues of problems in real-time. The real-time task recognition capabilities enable us to build a real-time updated operational dashboard for assembly work station analysis, which is presented in section 2 of this chapter.

The operational assembly work station analysis dashboard (OAWSAD) concept presented in section 2, aims to drive continuous improvement on the shop floor level. The dashboard provides the user with real-time performance information to evaluate his own performance. Furthermore the OAWSAD contains a number of charts and graphs which are frequently used in method study and lean manufacturing to optimize processes. These graphs are automatically generated and updated to provide instant insights on the effects of process changes. This significantly accelerates the PDCA improvement cycle often used on the shop floor in manufacturing companies. The customizable dashboard was created in an excel-file and linked to the python framework described in chapter 3.

The real-time monitoring and work station analysis framework presented in this chapter, could significantly change the way work study and continuous improvement of assembly line work stations are approached. On the level of the assembly station and assembly line, operators and production managers are enabled to analyze and rethink their by providing him with relevant real-time information and a very reactive feedback loop on both the work station and assembly line level respectively.

The proposed framework fundamentally changes the role of the industrial or methods engineer. Some of the tasks typically performed by the methods engineer, such as setting time standards and

generating work instructions, are (partially) taken over by the real-time monitoring system. These tasks are almost impossible to perform manually in current flexible high variety production environments. Instead, his main tasks now consist of disturbance handling and validating changes that originate from the shop floor level, as shown in Figure 83. This means that the engineer is now capable of managing multiple work stations/lines at once and react much faster. This reactivity is essential to stay competitive in today's market.

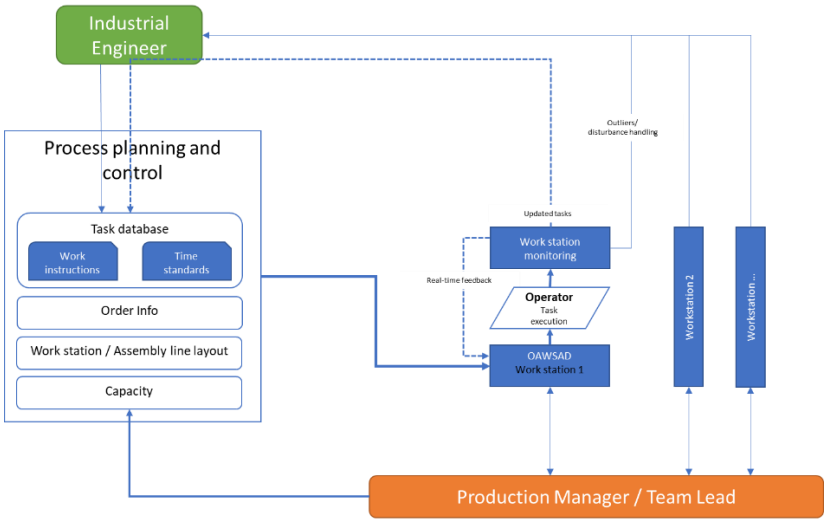



**Figure 83: Real-time monitoring data flow**

# Chapter 5 Conclusions and further research perspectives

Manufacturing industry is undergoing a clear trend towards mass customization. The ever increasing number of product variants to be produced, forces companies to make their processes more flexible and reactive, making them inherently more complex. In the specific case of mixed model assembly lines, flexibility is often still achieved through the use of human operators because of their capability to adapt and make their own decisions.

In order to stay competitive, flexible production environments need to be constantly monitored and re-evaluated. To do this, accurate and up-to-date information is required to provide insights in the performance of the system and support the decision-making process. In more automated environments, real-time monitoring systems based on for example RFID have already been used for some years, mainly in the fields of inventory management, production planning and quality management. However, to evaluate the performance of human in manual production environments, companies are still relying on traditional time and method study techniques. These techniques are mainly based on manual observations and are exorbitantly time-consuming. In this doctoral dissertation, we investigate the possibilities to use multi-camera based monitoring systems to evaluate the performance of mixed-model assembly work station operators and provide them with the necessary information to drive continuous improvement of their processes.

A number of reasons led to the choice to use camera-based systems. (1) In order to be accepted, an operator monitoring system should be as non-intrusive as possible. Existing tracking systems, such as motion capture suits, could hinder the person during his work and therefore these systems are typically not well received by the operators. (2) Video images are a rich source of information and have been used by

industrial engineers for many decades. Video images have an explanatory capability that is unmatched by any other motion capture sensor used today. Although they are not directly used by the proposed framework, the video images are a valuable by-product of the monitoring framework which can be used whenever deemed necessary. (3) Over the past decades, the price of cameras and data storage capacity has been constantly decreasing. Thereby it becomes possible to equip multiple work stations with camera systems that monitor the operator for longer time periods in a relatively inexpensive manner.

## 5.1. Research Contributions

In chapter 1, the framework for this research is being created. General trends in manufacturing industries and their challenges are being outlined. Through a literature review of the existing time study and operator monitoring techniques, we identified a gap between methods and the challenges of contemporary and future manufacturing systems and more specific, assembly systems. Furthermore, we elaborate on the role of human operators in these flexible environments and how novel (IoT) technology can be used to enhance human operator capabilities.

In chapter 2 we present a multi-camera based monitoring system that calculates the trajectory of the operator during his work. The video analysis method used, is based on the visual hull concept and was developed by IMEC. In the beginning of the chapter we describe the experimental setup and data sets used to perform and validate our research. Subsequently, although not developed in this research, we briefly describe the video analysis method to provide better understanding of the complete framework. In a third section of the chapter, we present a **data processing procedure** to clean up the raw output of the video analysis method and extract useful information from the raw trajectories. By linking the raw data to work station layout information, detailed information about the time distribution of the operator and the performance of the work station can be generated. The data framework proposed in this research, supports the methods engineer to automatically annotate captured video sequences and provide basic time measurements.

Chapter 3 presents a **work cycle clustering algorithm**. Unsupervised clustering of moving object trajectories has been of interest for many researchers. There are two main aspects that determine the accuracy of clustering frameworks: the distance measure and the classification procedure itself. An empirical study, based on a number of different experimental data sets, was performed to determine the most suitable combination of clustering method and similarity measure for this application. The proposed algorithm uses dynamic time warping to calculate the similarity between trajectories because of its' capability to deal with time deformations. The classification itself is based on normal hierarchical clustering methods. Hierarchical clustering however only provides insights in the similarity structure of a data set, therefore probabilistic permutation testing is used to automate the final classification. The proposed method reaches an average precision and accuracy that exceeds 90% based on the experimental data sets used in this research. This exceeds the performance of more commonly used classification methods.

Chapter 4 focuses on real-time operator monitoring and decision support. In order to provide the operator with real-time accurate and contextualized information, we need to be able to recognize his/her actions rapidly. Therefore we present a **real-time trajectory matching** method in the first section of the chapter. The method matches incoming trajectories to a set of models that were generated in the off-line classification analysis. Using the Keogh lower bound concept for dynamic time warping, the developed methodology only requires 0.07 seconds on average to process one frame in the video sequence. This is more than sufficient, knowing that experiments show that a rate of two frames per second still provides accurate classification results. The second section of this chapter focusses on real-time decision support on the shop floor level. In this section an **operational assembly work station analysis dashboard (OAWSAD)** is presented. Dashboards are being used more and more in manufacturing environments, however there is a clear lack of dashboards drive and accelerate continuous improvement on the work station or shop floor level. The OAWSAD presented in this section aims to bridge this gap

by integrating real-time performance data and automatically generated charts and diagrams that have proven their value in industrial engineering and continuous improvement for many years.

## 5.2. Limitations and Future research perspectives

The classification framework developed in this research is only validated on a set of experiments which don't really match the complexity of real-life assembly work stations. Changing regulations (GDPR) on the use of personal data of employees cause companies to be rather cautious about the use of video cameras on the shop floor. This eventually made experiments in real-life production environments more difficult than anticipated. However, we still feel that field test to validate the research results and prove the benefits for operators should be performed.

Another limitation is the speed of the video processing algorithms. The video analysis methods used in this research prove to be valuable for off-line analysis. However, real-time processing is not possible, even if the framerate of the cameras would be decreased to the minimum (2fps). Further research should therefore identify alternative means of location tracking, that provides the same accuracy and unobtrusiveness.

The methods presented in this research PhD thesis, remove the need for the industrial engineer to be physically present in the work station. This offers new perspectives for remote work station analysis and operator guidance. This concept is already in use for maintenance tasks, where maintenance engineers can offer support to field technicians through the use of wearable augmented reality devices such as google glasses. In further research projects we could translate these same concepts to the assembly environment. In recent years, we notice a trend towards more decentralized production facilities to increase the reactivity of the supply chain. Our framework enables industrial engineers to manage and support multiple assembly lines from a distance, as they are notified when disturbances arise and they are provided with the videos to perform further analysis.

The research presented in this doctoral thesis is not isolated and provides the first steps in a larger body of research projects to develop

production systems that are resistant to the challenges posed by the current market trends. Some of these research projects are described below.

**SBO-FLEXAS-VR** aims to develop a design framework for future flexible assembly work stations. Within this project, we aim to define and quantify assembly work station flexibility in order to adapt the work station design to the flexibility requirements. Human operators are still indispensable for many of these future assembly work stations, therefore suitable operator monitoring systems are a crucial part of the work station design to guarantee qualitive and efficient assembly processes.

As mentioned earlier, human operators are still by far the most flexible form of resources in the assembly system. In future flexible work stations the strengths of the human (creativity, problem solving ability, dexterity) are combined with the rapidly improving performance and affordability of robot systems with intuitive controls and context-aware information systems. This research thesis aligns well with this last aspect and is complementary to a number of ongoing research projects. **ICON Operator Knowledge** aims to develop a framework that is capable of capturing both implicit and explicit knowledge of experienced operators to improve the quality of assembly work instructions. To do this, a sensor-array will be developed to monitor human operators in their work station and provide information which can be used in the off-line creation and adaptation of assembly work instructions. Furthermore, a tool to automatically generate assemble sequences from the product CAD-files is being developed by researchers of UHasselt. This tool can suggest multiple assembly sequences, but does not take into account efficiency or quality aspects. On the other hand, the framework proposed is capable of automatically generating high-level work instructions based on the actual operations performed by the operator. Providing a link between these two methods, would enable us to automatically generate detailed work instructions, taking into account the feasibility and performance of the resulting assembly process.

ICON Operator Knowledge builds upon the results of another ICON project: **Operator Info.** The aim of this project is to provide assembly line operators with contextualized digital work instructions. Contextualized in this sense, means that the operator receives instructions which are adapted to his/her specific needs and the situation he/she is in. This concept of contextualized information is also used in **ADAPT**, an innovation project in collaboration with ARKITE. The aim of this project is to increase the adaptivity of the ARKITE HIM monitoring system and match the information provided by the guidance projector system to the specific profile of the operator. To do so, Bayesian belief networks will be used to estimate latent variables such as dexterity and the experience level of the operator. This information will be used to determine the profile of the operator. These Bayesian belief networks will mainly use time-based data as the input. The outcome of this PhD research will certainly help gather accurate manual assembly task times required to feed these Bayesian networks.

Real-time operator performance monitoring will also play a crucial role in the future **ICON project OperatorCapability.** There, the project goal is to develop a learning management system for manufacturing operations which allows to take up-to-date operator capability information into account for decision-making on operator-task allocation and operator on-the-job training. In this project we will develop a set of indicators which provide insights in the competences and capabilities of the operator. Operator capability is a dynamic concept however, operators learn but also lose knowledge and skills when these competences are not used for a longer period of time [162]. A generic model for an operator and workforce capability profile will be constructed based on physical competences, cognitive qualities, certifications, training results, performance stats, etc. In this context, the real-time operator performance monitoring system presented in this PhD thesis can provide valuable input to keep the work force capability matrix accurate and up-to-date and could therefore be integrated and extended in the framework developed in this project.

The OAWSAD presented in chapter 4 shows how real-time assembly process information can be used to support line balancing decisions. Keeping timing data of processes accurate and up-to-date is one of the main goals of the **ICON AssemblyBalance** project. The aim of this project is to develop more robust and dynamic methods to minimize productivity losses related to balancing issues of flexible assembly systems. This project was initiated by a number of assembly companies who identified a clear lack of supportive tools to stay within TAKT, level out imbalances and provide triggers to change the line balancing strategy. The OAWSAD already provides insights in the performance of line balancing strategies, however, the information shown in the dashboard could be used to automatically generate triggers and initiate actions when required.

In this research, we mainly start from the trajectories generated by the video analysis methods. However, the video images contain additional information that hasn't been used today. In the **ICON ErgoEyeHand**, the goal is to develop a real-time ergonomics assessment tool for assembly operators. The proposed framework uses motion capture suits to estimate the operators' posture and uses this information to automate the ergonomics risk assessment. However, the motion capture suits are very intrusive and often cause hindrance when used in real-life situations. The video analysis methods used in this research, provide 3D-models of the operator. This opens up perspectives to assess the posture of the operator based on the output of the video system without bothering the operator. The real-time ergonomics assessment tool could subsequently generate warnings or triggers when specific risks are detected. The OAWSAD could be extended with ergonomics information to create awareness amongst the operators and reduce the risk of long lasting injuries.

An overview of these projects is provided in Table 14.

### Table 14: overview ongoing and future linked research projects

| Project name | role | lead | status |
|---|---|---|---|
| **SBO Flexas-VR** | Design framework for flexible assembly work stations | UGent | Ongoing |
| **ICON Operator Knowledge** | Automatically generated and adapted work instructions based on operator feedback | Flanders Make | Ongoing |
| **ICON ErgoEyeHand** | Real-time ergonomics assessment and support | Flanders Make | Ongoing |
| **O&O ADAPT** | Operator-specific monitoring and guidance | ARKITE | Submitted |
| **ICON OperatorCapability** | Learning management systems for assembly operators | UGent | Submitted |
| **ICON AssemblyBalance** | Robust and reactive assembly line balancing | UGent | Submitted |

# References


1. Miller, D.M. and J.W. Schmidt, *Industrial Engineering and Operations Research*. 1985: Wiley.
2. Urwick, L., *Chapter 1 in handbook of Industrial Engineering*, in *Handbook of Industrial Engineering, 2nd Edition*. 1963, McGraw Hill: New York.
3. Kanawaty, G., *Introduction to work study*. 1995, Geneva: International Labour Organisation.
4. Rathod, A.S., R.G. Jadhav, and A.B. Babar, *An overview of Method Study and Study of Different Recording Techniques.* International Journal of Science and Research (IJSR) 2016. **5**(8): p. 1484-1491.
5. Groover, M.P., *Work Systems and the Methods, Measurement and Management of Work*. 2007, New Jersey: Pearson Prentice Hall.
6. Quick, J.H., J.H. Duncan, and J.A. Malcolm, *Work-factor time standards: measurement of manual and mental work*. 1962, New York: McGraw-Hill.
7. Freivalds, A., *Niebel's Methods, Standards and Work Design*. 2009, New York: McGraw-Hill.
8. Barnes, R.M., *Motion and time study*. 1940, New York: John Wiley & Sons.
9. Aft, L.S., *Work Measurement and Methods Improvement*. 2000, New York: John Wiley & Sons.
10. *Motion Economy in Lean Manufacturing*. Strategos.
11. Drira, A., H. Pierreval, and S. Hajri-Gabouj, *Facility layout problems: A survey.* Annual Reviews in Control, 2007. **31**(2): p. 255-267.
12. Meyers, F.E. and J.R. Stewart, *Motion and time study for lean manufacturing*. 2002, New Jersey: Prentice Hall.
13. Babbage, C., *On the economy of Machinery and Manufactures*. 1832, Cambridge: Cabridge University Press.
14. Taylor, F.W., *The Principles of Scientific Management*. 1911, New York: Harper Bros.
15. *CASE FILES: HENRY FORD*. The Franklin Institute.

16. Mundel, M.E., *Systematic Motion and Time Study*. 1947, New Jersey: Prentice Hall.
17. Wacker, J.G. and C. Sheu, *Effectiveness of Manufacturing Planning and Control systems on Manufacturing competitiveness: Evidence from Global manufacturing data.* International journal on Production Researck, 2006. **44**(5): p. 1015-1036.
18. Hopp, W.J. and M.L. Spearman, *Factory Physics*. Vol. 2. 2008, New York: McGraw-Hill.
19. Chae, B.K., et al., *The impact of advanced analytics and data accuracy on operational performance: A contingent resource based theory (RBT) perspective.* Decision Support Systems, 2014. **59**: p. 119-126.
20. Jumble, E., R. Haft, and M. Umble, *Enterprise resource planning: Implementation procedures and critical success factors.* European Journal of Operational Research, 2003: p. 241-257.
21. Haug, A., J.S. Arlbjorn, and A. Pedersen, *A classification model of ERP system data quality.* Industrial Management & Data systems, 2009. **109**(8): p. 1053-1068.
22. Kuhlang, P., et al., *Morphology of time data management - systematic design of time data management processes as fundamental challenge in industrial engineering.* Industrial and Systems Engineering, 2014. **16**(4): p. 415-432.
23. ifaa, *ifaa-trendbarometer*. 2012, ifaa.
24. Petzelt, D.J., *Konzept zur rechnergestützten Bestimmung von Sollzeiten in der Montageplannung*. 2010, Aachen: Shaker.
25. Boothroyd, G., P. Dewhurst, and W. Knight, *Product Design for Manufacture and Assembly*. 2011, Boca Raton: CRC Press, Taylor and Francis group.
26. Schaub, K., et al., *The European Assembly Worksheet.* Theoretical Issues in Ergonomics Science, 2010. **14**(6): p. 1-23.
27. Almström, P. and A. Kinnander, *The productivity potential assessment method: Assessing and benchmarking the improvement potential in manufacturing systems at shop-floor level.* International Journal of Productivity and Performance Management, 2011. **60**(7): p. 758-770.
28. Kuhlang, P., et al., *Morphology of time data management – systematic design of time data management processes as*

*fundamental challenge in industrial engineering.* Int. J. Industrial and Systems Engineering, 2014. **16**(4): p. 415-432.
29. ElMaraghy, H. and W. ElMaraghy. *Smat Adaptable Assembly Systems*. in *Procedia CIRP*. 2016.
30. Best, T., *Work measurement in skilled labor environments*. 2012.
31. Nadler, G., *Standard Data problem and solutions.* American Society of Mechanical Engineers, 1953: p. 9.
32. Konz, S., *Time study*, in *Handbook of Industrial Engineering*. 2001, John Wiley & Sons: New York. p. 1391-1408.
33. Lindemeyer, C.R., *using multiple regression for work measurement & standard data applications*. 1993, Pendleton: c4 Productivity Press.
34. Karger, D.W. and F.H. Bayha, *Engineered work measurement*. 1957, New York: Industrial Press.
35. Zandin, K.B., *MOST work measurement systems*. 2003, New York: Marcel Dekker.
36. Niebel, B.W. and A. Freivalds, *Methods, Standards and Work Design*. 2003, Boston: McGraw-Hill.
37. Laubrass [cited 2019 May 5th]; Available from: https://www.laubrass.com/umtplus.
38. ACSCO. *Timer Pro*. [cited 2019 May 6th]; Available from: https://www.acsco.com/.
39. Corwin, S., *Picturing Efficiency: Precisionism, Scientific Management, and the Effacement of Labor*, in *Representations*. 2003, University of California Press.
40. Beliveau, A., *The micromotions Films of Frank and Lilian Gilbreth*. 2010, History of Science Society.
41. Baudin, M., *Using videos to improve operations | Part 1 – Overview and Motivation*. 2013.
42. Konz, S., *Methods Engineering*, in *Handbook of Industrial Engineering*. 2001, John Wiley & Sons: New York. p. 1353-1390.
43. Karger, D.W. and W.M. Hancock, *Advances Work Measurement*. 1982, New York: Industrial Press.
44. Taylor, P., *From figure skaters to the factory floor*. 2011, Financial Times.
45. Lukac, D. *The fourth ICT-based industrial revolution "Industry 4.0" — HMI and the case of CAE/CAD innovation with EPLAN*

*P8*. in *23rd Telecommunications Forum Telfor (TELFOR)*. 2015. Belgrade.
46. Duüren, O.v. *The Industrial Revolution 4.0*. [cited 2019 May 26th]; Available from: https://www.thedualarity.com/industrial-revolution-4-0-2/.
47. IDC, *"Industrie 4.0 in Deutschland - Startschuss fur die ITK-basierte vierte industrielle Revolution"*. 2014.
48. Monostori, L., et al., *Cyber-physical systems in manufacturing.* CIRP Annuals, 2016. **65**(2): p. 621-641.
49. Monaco, D. *What is explainable AI and why is it important*. 2019.
50. Chander, A. and R. Srinivasan, *Evaluating Explanations by Cognitive Value*, in *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*. 2018. p. 314-328.
51. Van Leeckwyck, F., *Das I4.0 Portal*. 2017.
52. Ruppert, T., et al., *Enabling Technologies for Operator 4.0: A Survey.* Applied Sciences, 2018. **8**(9).
53. Romero, D., et al., *The Operator 4.0: Human Cyber-Physical Systems & Adaptive Automation Towards Human-Automation Symbiosis Work Systems.* Advances in Production Management Systems, 2016. **10**: p. 677_686.
54. Chen, H., R.H. Chiang, and V.C. Storey, *Business intelligence and analytics: from big data to big impact.* MIS Quarterly., 2012. **36**(4): p. 1165-1188.
55. Wang, S., J. Wan, and D. Li, *Implementing smart factory of indusrie 4.0: an outlook.* International journal of distributed sensor networks, 2016. **12**: p. 1-12.
56. Ruppert, T. and J. Abonyi, *Software Sensor for Activity-Time Monitoring and Fault Detection in Production Lines.* Sensors, 2018. **18**(7).
57. Krahnstoever, N., et al. *Activity Recognition using Visual Tracking and RFID*. in *Proceedings of the 7th IEEE Workshop on applications of computer vision*. 2005. Breckenridge.
58. Sardroud, J.M., *Influence of RFID technology on automated management of construction materials and components.* Scientia Iranica, 2012. **19**(3): p. 381-392.
59. Arkan, I. and H. Van Landeghem, *Evaluating the performance of a discrete manufacturing process using RFID: a case study.*

Robotics and computer-aided manufacturing, 2013. **29**(6): p. 502-512.

60. Gladysz, B., K. Santarek, and C. Lysiak. *Dynamic Spaghetti diagrams. A case study of Pilot RTLS implementation*. in *Intelligent Systems in Production Engineering and Maintenance – ISPEM 2017. ISPEM 2017*. 2017. Wroclaw.
61. Koskimaki, H., et al. *Activity recognition using wrist-worn inertial measurement unit: a case study for industrial assemby lines*. in *17th Mediterrenean Conference on Control and Automation*. 2009. Thessaloniki.
62. Ward, J.A., et al., *Activity Recognition of Assembly Tasks Using Body-Worn Microphones and Accelerometers.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006. **28**(10): p. 1553-1567.
63. Stiefmeier, T., et al., *Wearable activity tracking in car manufacturing.* IEEE Pervasive computing, 2008. **7**(2): p. 42-50.
64. Wang, Z., et al., *Stereo-vision based hybris manufacturing process for precision metal parts.* Precision engineering, 2015. **42**: p. 1-5.
65. Michalos, G., et al., *Robot Path Correction Using Stereo Vision System.* Precedia CIRP, 2012. **3**: p. 352-357.
66. Fleck, S. and W. Strasser. *Adaptive Probabilistic Tracking Embedded in a Smart Camera.* in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. 2005. San Diego.
67. Kaber, D.B., et al., *Situation Awareness implications of adaptive automation for informatio processing in an air traffic control-related task.* International journal of Industrial Ergonomics, 2006. **16**: p. 447-462.
68. Wright, I., *Engineering.com*. 2017.
69. Shafiq, S.I., et al., *Virtual Engineering Factory: Creating Experience Base for Industry 4.0.* Cybernetics and Systems, 2016. **47**: p. 32-47.
70. Bauer, W., et al., *Digitalization of industrial value chains - a review and evaluation of existing use cases of industry 4.0 in Germany.* LogForum - scientific journal of logistics, 2018. **14**(3): p. 331-340.
71. Zheng, Y., *Trajectory data mining: An overview.* ACM Transactions on intelligent systems technology, 2015. **6**(3).

72. Schmitt, P., F. Thiesse, and E. Fleisch. *Adoption and diffusion of RFID technology in the automotive industry*. in *15th European Conference on Information Systems*. 2008. Sankt-Gallen.
73. Huang, G.Q., Y.F. Zhag, and P.Y. Jiang, *RFID-based wireless manufacturing for real-time management of job shop WIP inventories.* International Journal of Advanced Manufacturing Technology, 2008. **36**: p. 752-764.
74. Karkkainen, M., and J. Holmstrom, *Wireless product identification: enabler for handling efficiency , customisation and information sharing.* Supply Chain Management: An International Journal, 2002. **7**: p. 242-252.
75. Chappel, G., L. Ginsburg, P. Schmidt, J. Smith, and J. Tobolski, *Auto-ID on the Line: The Value of Auto-ID Technology in Manufacturing*. 2003, Auto-ID Center Massachusetts Institute of Technology: Massachusetts.
76. Morris, B.T. and M.M. Trivedi, *Trajectory learning for activity understanding: Unsupervised, Multilevel, and Long-Term adaptive approach.* IEEE Transactions on pattern analysis and machine intelligence, 2011. **33**(11): p. 2287-2302.
77. Bauters, K., et al. *Multicamera complexity assessment system for assembly line work stations*. in *The European Concurrent Engineering Conference, Proceedings.* 2013. Lincoln.
78. Laurentini, A., *The visual hull concept for silhouette-based image understanding.* IEEE transactions on Pattern Analysis and Machine Intelligence, 1994. **16**(2): p. 150-162.
79. Kim, H., et al. *Compensated visual hull with GPU-based optimization*. in *PCM '08 Proceedings of the 9th Pacific Rim Conference on Multimedia: Advances in Multimedia Information Processing*. 2008. Tainan - Taiwan.
80. Liang, C. and K.-y.K. Wong, *3D reconstruction using silhouettes from unordered viewpoints.* Journal of Image and vision computing, 2010. **28**(4): p. 579-589.
81. Gutierrez, A., et al. *Persistent Homology for 3D Reconstruction Evaluation*. in *Proceedings of the 4th international conference on Computational Topology in Image Context*. 2012. Bertinoro.
82. Slembrouck, M., et al. *High performace multi-camera tracking using shapes-from-silhouettes and occlusion removal*. in

*Proceedings of the 9th International Conference on Distributed Smart Cameras*. 2015. New York.

83. Hastie, T., R. Tibshirani, and J. Friedman, *Kernel Smoothing Methods*, in *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2009, Springer: New York. p. 191-218.
84. Alvares, L.O., et al. *A Model for Enriching Trajectories with Semantic Geographical Information*. in *15th ACM International Symposium on Geographic Information Systems*. 2007. Seattle.
85. Rokach, L. and O. Maimon, *Clustering Methods.*, in *Data Mining and Knowledge Discovery Handbook.* 2005, Springer: Boston.
86. Yuan, G., et al., *A review of moving object trajectory clustering.* Artificial Intelligence Review, 2017. **47**: p. 123-144.
87. Fu, Z., H. Wu, and T. Tan. *A review of moving object trajectory clustering*. in *Proceedings of the International Conference on Image processing (ICIP 05)*. 2005.
88. Bashir, F.I., A.A. Khokhar, and D. Schonfeld. *Segmented trajectory based indexing and retrieval of video data*. in *International Conference on Image Processing (ICIP '03)*. 2003.
89. Huttenlocher, D., G. Kladerman, and W. Rucklidge, *Comparing Images Using the Hausdorff Distance.* IEEE TRANSACTIONSON P A m R N ANALYSIS AND MACHINE INTELLIGENCE, 1993. **15**(9): p. 850-863.
90. Chen, J.Y., et al. *Clutering of trajectories based on Hausdorff distance*. in *Proceedings of the 2011 international conference on electronics, communications and control*. 2011.
91. Wei, L.X., et al., *Trajectory classification based on Hausdorff distance and longest.* International journal of electronics and information technology, 2013. **35**(4): p. 784-790.
92. Lin, B. and J. Su, *One Way Distance: For Shape Based Similarity Search.* Geoinformatica, 2008. **12**: p. 117-142.
93. Vlachos, M., G. Kollios, and D. Gunopulos. *Discovering similar multidimensional trajectories*. in *Proceedings of the 18th international conference on data engineering*. 2002. Bali.
94. Buzan, D., S. Sclaroff, and G. Kollios. *Extraction and clustering of Motion Trajectories in Video*. in *Proceedings of the 17th international conference on Pattern Recognition*. 2004. Cambridge.

95. Liu, H. and M. Schneider. *Similarity Measurement of Moving Object Trajectories*. in *Proceedings of the 3rd ACM SIGSPATIAL International workshop on geostreaming*. 2012. California.
96. Besse, P.C., et al., *Review and Perspective for distance-baed clusteirng of vehicle trajectories.* IEEE Transactions on intelligent transportation Systems, 2016. **17**(11): p. 3306-3317.
97. Eiter , T. and H. Manilla, *Calculating discrete Fréchet Distance*. 1994.
98. Müller, M., *Dynamic Time Warping.* Information Retrieval for music and motion, 2007: p. 69-84.
99. Vaughan, N. and B. Gabrys. *Comparing and combining time series trajectories using dynamic time waring*. in *Proceedings of the 20th international conference on knowledge based and intelligent information and engineering systems*. 2016. YORK.
100. Chen, P., et al., *A Dynamic Time Warping based Algorithm for Trajectory Matching in LBS.* International Journal if Database Theory and Application, 2013. **6**(3): p. 39-48.
101. Pohl, H. and A. Hadjakos, *Dance pattern recognition using dynamic time warping*. 2010.
102. Blackburn, J. and E. Ribeiro. *Human motion recognition using Isomap and dynamic time warping*. in *Proceedings of the 2nd conference on Human motion: understanding, modeling, capture and animation*. 2007. Rio de Janeiro.
103. Petitjean, F., A. Ketterlin, and P. Gançarski, *A Global averagnig method for dynamic time warping, with applications to clustering.* Pattern Recognition, 2011. **44**(3): p. 478-693.
104. AL-Naymat, G., S. Chawla, and J. Taheri, *SparseDTW: A Novel Approach to Speed up Dynamic Time Warping.* The 2009 Australasian Data Mining, 2012. **101**: p. 117-127.
105. Goyal, H., et al., *Normalization of Data in Data mining.* International Journal of Software and Web Sciences, 2014. **14**: p. 32-33.
106. Farley, C. and A.E. Raftery, *"How Many Clusters? Which Clustering Method?* 1998, Department of Statistics University of Washington: Washington DC.
107. Han, J., M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*. 2011, Waltham: Morgan Kaufmann.
108. Hartigan, J.A., *Clustering algorithms*. 1975: Wiley.

109. Kaufmann, L. and P. Rousseeuw, *Clustering by mean of medoids*, in *Data Analysis based on the L1-Norm and Related Methods.* 1987. p. 405-416.
110. Ayramo, S. and T. Karkkainen, *Introduction to partitioning-based clustering methods with a robust example*. 2006.
111. Dhillon, I. and D. Modha, *Concept Decompositions for Large Sparse Text Data Using Clustering.* Machine Learning, 2001. **42**(1-2): p. 143–175.
112. Shindler, M., A. Wong, and A. Meyerson. *Fast and accurate k-means for large datasets*. in *Proceedings of the 24th international conference on Neural information processing systems*. 2011. Granada.
113. Kaufman, L. and P. Rousseeuw, *Finding groups in data: an introduction to clustering analysis*. 1990: Wiley.
114. Sneath, P.H.A. and R.R. Sokal, *Numerical taxonomy: the principles and practice on numerical classification*. 1973, San Francisco: W. H. Freeman and Co.
115. King, B., *Step-wise clustering procedures.* Journal of the American statistics association, 1967. **69**: p. 68-101.
116. Murtagh, F., *A survey of recet advances in hierarchical clustering algorithms.* The computer journal, 1983. **26**(4): p. 354-359.
117. Ward, J.H., *Hierarchical grouping to optimize an objective function.* Journal of the American Statistics Association, 1963. **58**: p. 236-244.
118. Ester, M., et al. *A Density-Based Algorithm for Discovering Clusters in Large Spatioal Databases with Noise*. in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. 1996. Portland, Oregon.
119. Ankerst, M., et al. *OPTICS: ordering points to identify the clustering structure*. in *SIGMOD '99 Proceedings of the 1999 ACM SIGMOD international conference on Management of data*. 1999. Philadelphia.
120. Nanni, M. and D. Pedreschi, *Time-focused clustering of trajectories of moving objects.* Journal of Intelligent Information Systems, 2006. **27**(3): p. 267-289.
121. Mao, Y., et al., *An Adaptive Trajectory Clustering Method Based on Grid and Density in Mobile Pattern Analysis.* Sensors, 2017. **17**(9).

122. Buchin, K., et al., *Detecting commuting patterns by clustering subtrajectories.* International Journal of Computational Geometry and Applications, 2011. **21**(3): p. 253-282.
123. Pelekis, N., I. Kopanakis, and E.E. Kotsifakos, *Clustering uncertain trajectories.* Knowledge and Information Systems, 2011. **28**(1): p. 117–147.
124. Chen, J., et al. *Sketch-based uncertain trajectories clustering.* in *9th International Conference on Fuzzy Systems and Knowledge Discovery*. 2012.
125. Palma, A.T., et al. *A clustering-based approach for finding interesting places in trajectories*. in *Proceedings of the 2008 ACM Symposium on Applied Computing (SAC)*. 2008. Fortaleza.
126. Cai, G., K. Lee, and I. Lee, *Mining Mobility Patterns from Geotagged Photos Through Semantic Trajectory Clustering.* Cybernetics and Systems, 2018. **49**(4): p. 234-256.
127. Portugal, I., P. Alencar, and D. Cowan, *Developing a Spatial-Temporal Contextual and Semantic Trajectory Clustering Framework.* CoRR, 2017.
128. Yan, Z., et al., *Semantic Trajectories: Mobility Data Computation and Annotation.* ACM Transactions on Intelligent Systems and Technology, 2013. **4**(3).
129. Salvador, C. and P.K. Chan. *Determining the Number of Clusters/Segments in Hierarchical Clustering/Segmentation Algorithms*. in *ACM Transactions on Intelligent Systems and Technology*. 2013. Boca Raton.
130. Tibshirani, R., G. Walther, and T. Hastie, *Estimating the number of clusters in a data set via the gap statistic.* journal of statistical software, 2001. **63**(2): p. 411-423.
131. Breiman, L., et al., *Classification and regression trees*. 1984: Chapman & hall.
132. Bruzesse, D. and D. Vistocco, *DESPOTA: dendrogram slicing through a permutation test approach.* journal of classification, 2015. **32**(2): p. 283-304.
133. Singh, D., *A Study on the Use of Non-Parametric Tests for Experimentation with cluster analysis.* International Journal of Engineering and Management Research, 2013. **3**(6): p. 64-72.
134. *ttable.org*.

135. Keogh, E. and C.A. Ratanamahatana, *Exact indexing of dynamic time warping.* Knowledge and information systems, 2005. **7**(3): p. 358-386.
136. Capitani, P. and P. Ciaccia, *Warping the time on data streams.* Data & knowledge engineering, 2006. **62**: p. 438-458.
137. Rath, T. and R. Manmatha, *Lower-bounding of dynamic time warping distances for multivariate time series*. 2002: University of Massachusetts.
138. Henri, T., et al. *Designing manufacturing dashboards on the basis of a key performance indicator survey*. in *9th CIRP Conference on Manufacturing Systems*. 2016. Guimaraes.
139. Gude, A. *How a performance dashboard can help you achieve continuous improvement*. [cited 2019 May 27th]; Available from: https://www.bdc.ca/en/blog/pages/achieve-continuous-improvement-with-performance-dash.aspx.
140. Yusof, E.M.M. and M.S. Othman, *A Review on the dashboard characteristics for manufacturing organizations.* Journal of information systems research and innovation, 2012: p. 29-34.
141. Realyvasquez-Vargas, A., et al., *Applying the Plan-Do-Check-Act (PDCA) Cycle to Reduce the Defects in the Manufacturing Industry. A Case Study.* Applied Scieces, 2018. **8**(11).
142. Zahid, H. and H. Zanowar, *Improvement of Effectiveness by Applying PDCA Cycle or Kaizen: An experimental study on Engineering Students.* Journal of scientific research, 2018. **10**(2): p. 159-173.
143. Christoph, G., et al. *The operational process dashboard for manufacturing*. in *CIRP Conference on manufacturing Systems*. 2013. Elsevier.
144. Poobalan, P., *Equipment and Process Performance Improvement Using Automated Monitoring System*. 2009, Universiti Teknologi Malaysia (UTM): Malaysia.
145. Higgins, P., et al., *An operations Model for the Extended Manufacturing Enterprise*, in *11th Conference on Concurrent Enterprising*. 2005: Munich.
146. Bai, A., et al., *Remote Production Dashboard: A real-time and online Control Technology for discrete manufacturing based on knowledge service*, in *IEEE 17Th International Conference on Industrial Engineering and Engineering Management*. 2010: Xiamen.

147. Nadoveza, D. and D. Kiritsis, *Concept for context-aware Manufacturing Dashboard Applications*, in *IFAC Conference on Manufacturing Modelling, Management and Control*. 2013: Saint Petersburg. p. 204-209.
148. Mazumdar, S., et al., *A Knowledge Dashboard for Manufacturing Industries*, in *ESWC: Extended Semantic Web Workshop*, R. García-Castro, D. Fensel, and G. Antoniou, Editors. 2011: Heraklion, Greece.
149. Kapp, R., R. Weimer, and E. Westkämper, *Factory cockpit: A factory simulator for the entire factory life cycle*, in *International Seminar on Manufacturing Systems*. 2006: Ljubljana. p. 203-208.
150. Shamsuzzoha, A., et al., *Dashboard User Interface for Measuring Performance Metrics: Concept from Virtual Factory Approach*, in *International conference on Industrial engineering and Operations Management*. 2014.
151. *Smart shopfloor*. [cited 2019 May 27]; Available from: https://smart-shopfloor.com/en/home/.
152. *Manufacturing Dashboard*. [cited 2019 May 27th]; Available from: https://www.simplekpi.com/KPI-Dashboard-Examples/Manufacturing-Dashboard-Example.
153. *Achieve Real-time Enterprise Manufacturing Intelligence to Improve Visibility, Synchronization and Control*. [cited 2019 May 27th]; Available from: http://www.apriso.com/products/manufacturing_process_intelligence/.
154. *redlion*. 2018, red lion controls.
155. Adnan, A.N., et al., *Implementation of Continuos Flow System in Manufacturing Operation.* Applied Mechanics and Materials, 2014. **393**: p. 9-14.
156. Teixeira, E.L.S., *Statistical Process Control Application in Automotive Industru*, in *ABCM International congres of Mechanical Engineering*. 2017: Curitiba, Brazil.
157. Hys, K. and A. Domagala, *Application of spaghetti chart for production process streamlining. Case study.* Archives of Material Science and Engineering, 2018. **89**(2): p. 64-71.
158. Senderska, K., A. Mares, and S. Vaclav, *Spaghetti diagram application wor workers' movement analysis.* U.P.B. Scientific Bulletin, 2017. **79**(1): p. 139-150.

159. *All about spaghetti diagrams*. 2015 [cited 2019 May 28th]; Available from: https://www.allaboutlean.com/spaghetti-diagrams/.
160. Driscoll, D. and D. Thilakawardana, *The Definition of assmbly line balancing difficulty and evaluation of balance solution quality.* Robotics and computer-aided manufacturing, 2001. **17**(1-2): p. 81-86.
161. Bragg, S., *accountingtools*. 2018.
162. Hoedt, S., et al., *Countering the forgetting effect in mixed-model manual assembly.* , in *IFAC Symposium on Information Control Problems in Manufacturing (INCOM)* 2018, Elsevier: Bergamo.

# Appendix 1

## WARD_LINKAGE

| | LCSS | | | | Frechet | | | | Dynamic Time Warping | | | | One-Way Distance | | | | Hausdorff | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **SEQUENCE** | **RI** | **Precision** | **Recall** | **F1** | **RI** | **Precision** | **Recall** | **F1** | **RI** | **Precision** | **Recall** | **F1** | **RI** | **Precision** | **Recall** | **F1** | **RI** | **Precision** | **Recall** | **F1** |
| seq01 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq02 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq03 | 0.771429 | 0.965517 | 0.717949 | 0.823529 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.5428 | 0.8571 | 0.4615 | 0.6 | 0.6666 | 0.9387 | 0.5897 | 0.7244 |
| seq04 | 0.6373 | 0.9411 | 0.6153 | 0.7441 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq05 | 0.6969 | 0.9487 | 0.6727 | 0.7872 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.5909 | 0.9117 | 0.5636 | 0.6966 |
| seq06 | 0.4857 | 0.8363 | 0.5055 | 0.6301 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.8666 | 0.9873 | 0.8571 | 0.9176 |
| seq07 | 0.5523 | 0.6999 | 0.2545 | 0.3733 | 0.7047 | 0.9615 | 0.4545 | 0.6172 | 0.7619 | 0.9687 | 0.5636 | 0.7126 | 0.6571 | 0.8064 | 0.4545 | 0.5813 | 0.5809 | 0.7619 | 0.2909 | 0.421 |
| seq08 | 0.5055 | 0.8837 | 0.4871 | 0.628 | 0.8571 | 0.985 | 0.8461 | 0.9103 | 1 | 1 | 1 | 1 | 0.5604 | 0.913 | 0.5384 | 0.6775 | 0.5604 | 0.913 | 0.5384 | 0.6774 |
| seq09 | 0.5055 | 0.7083 | 0.309 | 0.4303 | 0.7252 | 0.9687 | 0.5636 | 0.7126 | 0.6593 | 0.9615 | 0.4545 | 0.6172 | 0.6043 | 0.8064 | 0.4545 | 0.5813 | 0.4835 | 0.6818 | 0.2727 | 0.3896 |
| seq10 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq11 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **AVERAGE** | **0.74133** | **0.907592** | **0.687459** | **0.765139** | **0.935182** | **0.992291** | **0.896745** | **0.930918** | **0.947382** | **0.993655** | **0.910736** | **0.939073** | **0.851327** | **0.9439** | **0.8099** | **0.858191** | **0.795355** | **0.926764** | **0.737491** | **0.802418** |

## Single linkage

| | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| seq01 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq02 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq03 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.6666 | 0.8208 | 0.7051 | 0.7586 |
| seq04 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.7362 | 0.8461 | 0.8461 | 0.8461 |
| seq05 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.6969 | 0.8181 | 0.8181 | 0.8181 |
| seq06 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.6571 | 0.8481 | 0.7362 | 0.7882 |
| seq07 | 0.7238 | 0.7826 | 0.6545 | 0.7128 | 0.8952 | 0.9782 | 0.8181 | 0.891 | 1 | 1 | 1 | 1 | 0.8095 | 0.8181 | 0.8181 | 0.8181 | 0.6476 | 0.7368 | 0.509 | 0.6021 |
| seq08 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.7326 | 0.8461 | 0.8461 | 0.8461 | 0.7362 | 0.8461 | 0.8461 | 0.8461 |
| seq09 | 0.5824 | 0.6545 | 0.6545 | 0.6545 | 1 | 1 | 1 | 1 | 0.7802 | 0.8181 | 0.8181 | 0.8181 | 0.7802 | 0.8181 | 0.8181 | 0.8181 | 0.7802 | 0.8181 | 0.8181 | 0.8181 |
| seq10 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq11 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **AVERAGE** | **0.936927** | **0.948827** | **0.937182** | **0.942482** | **0.990473** | **0.998018** | **0.983464** | **0.990091** | **0.980018** | **0.983464** | **0.983464** | **0.983464** | **0.938391** | **0.952936** | **0.952936** | **0.952936** | **0.810982** | **0.884918** | **0.843518** | **0.861573** |

## Average linkage

| | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| seq01 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq02 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq03 | 0.8761 | 0.9851 | 0.8461 | 0.9103 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.5428 | 0.8571 | 0.4615 | 0.6 | 0.6666 | 0.9387 | 0.5897 | 0.7244 |
| seq04 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq05 | 0.6969 | 0.8181 | 0.8181 | 0.8181 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq06 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.8666 | 0.9873 | 0.8571 | 0.9176 |
| seq07 | 0.6571 | 0.7435 | 0.5272 | 0.617 | 0.8952 | 0.9782 | 0.8181 | 0.891 | 1 | 1 | 1 | 1 | 0.6571 | 0.8064 | 0.4545 | 0.5813 | 0.5619 | 0.6956 | 0.2909 | 0.4103 |
| seq08 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.5604 | 0.913 | 0.5384 | 0.6775 | 0.7362 | 0.8461 | 0.8461 | 0.8461 |
| seq09 | 0.6813 | 0.7826 | 0.6545 | 0.7128 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.6043 | 0.8064 | 0.4545 | 0.5813 | 0.5494 | 0.7692 | 0.3636 | 0.4938 |
| seq10 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq11 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **AVERAGE** | **0.901036** | **0.939027** | **0.895082** | **0.914382** | **0.990473** | **0.998018** | **0.983464** | **0.990091** | **1** | **1** | **1** | **1** | **0.851327** | **0.9439** | **0.8099** | **0.858191** | **0.852791** | **0.930627** | **0.8134** | **0.853836** |

## COMPLETE linkage

| | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| seq01 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq02 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq03 | 0.7714 | 0.9655 | 0.7179 | 0.8235 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.6666 | 0.9387 | 0.5897 | 0.7244 |
| seq04 | 0.6373 | 0.9411 | 0.6153 | 0.7441 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq05 | 0.6969 | 0.9487 | 0.6727 | 0.7872 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq06 | 0.4666 | 0.8431 | 0.4725 | 0.6056 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0.8666 | 0.9873 | 0.8571 | 0.9176 |
| seq07 | 0.5333 | 0.6153 | 0.2909 | 0.395 | 0.7619 | 0.9687 | 0.5636 | 0.7126 | 0.7047 | 0.9615 | 0.4545 | 0.6172 | 0.8095 | 0.8181 | 0.8181 | 0.8181 | 0.5142 | 0.5999 | 0.2181 | 0.3199 |
| seq08 | 0.8571 | 0.985 | 0.8461 | 0.9103 | 0.8571 | 0.985 | 0.8461 | 0.9103 | 1 | 1 | 1 | 1 | 0.5604 | 0.913 | 0.5384 | 0.6774 | 0.5054 | 0.8837 | 0.4871 | 0.628 |
| seq09 | 0.5054 | 0.7083 | 0.309 | 0.4303 | 1 | 1 | 1 | 1 | 0.8791 | 0.9782 | 0.8181 | 0.8911 | 0.8791 | 0.9782 | 0.8181 | 0.891 | 0.5054 | 0.7083 | 0.309 | 0.4303 |
| seq10 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| seq11 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| **AVERAGE** | **0.769818** | **0.909727** | **0.7204** | **0.790545** | **0.965364** | **0.995791** | **0.946336** | **0.965718** | **0.962164** | **0.994518** | **0.933873** | **0.9553** | **0.931727** | **0.973573** | **0.924964** | **0.944227** | **0.823473** | **0.919809** | **0.769182** | **0.820018** |

# Appendix 2

One Way Distance

```
def OWD(traj1, traj2):
    ts1 = traj1.route
    ts2 = traj2.route

    a = [Dpoint(ts1[i], ts2) for i in range(len(ts1))]
    b = [Dpoint(ts2[i], ts1) for i in range(len(ts2))]


    Dowd12 = sum(a)/traj1.distance()
    Dowd21 = sum(b)/traj2.distance()
    owd = (Dowd12 + Dowd21)/2
    return owd
```

Hausdorff distance

```
def hausdorff(traj1, traj2):
    h = 0

    ts1 = traj1.route
    ts2 = traj2.route
    for a in ts1:
        shortest = 1000000000000000000
        for b in ts2:
            d = dist(a, b)
            if d<shortest:
                shortest = d
        if shortest > h:
            h = shortest
    return h
```

Longest common subsequence

```
def LCSS(traj1, traj2, e, d):
    l1 = len(traj1.route)
    l2 = len(traj2.route)
    C = [[0 for j in range(l2)] for i in range(l1)]

    for i in range(1, l1):
```

```
        for j in range(1, l2):
            a = traj1.route[i]
            b = traj2.route[j]

            if abs(a[0]-b[0])<e and abs(a[1]-b[1])<e and
abs(i-j)<d:
                C[i][j]=C[i-1][j-1] + 1
            else:
                 C[i][j] = max(C[i-1][j], C[i][j-1])


    return 1-float(C[-1][-1])/min(l1-1,l2-1)




Dynamic time warping

def dist(l1, l2):
    #calculates the distance between two points in a
route
    d = math.sqrt((l1[0]-l2[0])**2 + (l1[1]-l2[1])**2)
    return d

def DTWdist(traj1, traj2):
    #calculates the DTW distance between two time series


    numrows = len(traj1.route)
    numcolumns = len(traj2.route)


    matrix = [[0 for j in range(numcolumns)] for i in
range(numrows)]

    for i in range(numrows):
        for j in range(numcolumns):
            d = dist(traj1.route[i], traj2.route[j])/100
            matrix[i][j] = d

    DTW = [[0 for j in range(numcolumns)] for i in
range(numrows)]

    #initialize rows and colums
    DTW[0][0] = matrix[0][0]
```

```
    #first row
    for i in range(1,numrows):
        DTW[i][0] = DTW[i-1][0] + matrix[i][0]

    #first column
    for j in range(1,numcolumns):
        DTW[0][j] = DTW[0][j-1] + matrix[0][j]

    # rest of the matrix

    for i in range(1, numrows):
        for j in range(1, numcolumns):
            choices = DTW[i-1][j], DTW[i][j-1], DTW[i-
1][j-1]
            DTW[i][j] = min(choices) + matrix[i][j]



    #mprint(DTW)

    distance = DTW[-1][-1]




    #calculate warping path
    #init
    i = numrows-1
    j = numcolumns-1
    wp = [(i, j)]

    while (i>0 and j>0):
        choices = DTW[i-1][j], DTW[i-1][j-1], DTW[i][j-
1]
        val, idx = min((val, idx) for (idx, val) in
enumerate(choices))
        if idx==0:
            wp.append((i-1, j))
            i -=1

        elif idx==1:
            wp.append((i-1, j-1))
            i -= 1
            j -= 1
```

```
        elif idx==2:
            wp.append((i, j-1))
            j -=1

        else:
            print "something went terribly wrong ☹,
check the DTW traceback part of the DTW algorithm"


    if(i>0):
        for k in reversed(range(i)):
            wp.append((k, 0))
    elif(j>0):
        for l in reversed(range(j)):
            wp.append((0, l))

    wp.reverse()



    norm = len(wp)
   # print "DTW distance "+str(distance)
    return distance/norm, wp
```

Dynamic time warping (recursive calculation – used in real-time framework)

```
ef euc_dist(pt1,pt2):
    return math.sqrt((pt2[0]-pt1[0])*(pt2[0]-
pt1[0])+(pt2[1]-pt1[1])*(pt2[1]-pt1[1]))

def _c(ca,i,j,P,Q):
    if ca[i,j] > -1:
        return ca[i,j]
    elif i == 0 and j == 0:
        ca[i,j] = euc_dist(P[0],Q[0])
    elif i > 0 and j == 0:
        ca[i,j] = max(_c(ca,i-
1,0,P,Q),euc_dist(P[i],Q[0]))
    elif i == 0 and j > 0:
        ca[i,j] = max(_c(ca,0,j-
1,P,Q),euc_dist(P[0],Q[j]))
    elif i > 0 and j > 0:
        ca[i,j] = max(min(_c(ca,i-1,j,P,Q),_c(ca,i-1,j-
1,P,Q),_c(ca,i,j-1,P,Q)),euc_dist(P[i],Q[j]))
```

```
    else:
        ca[i,j] = float("inf")

    return ca[i,j]


def _dtw(traj1, traj2):

    x1, y1, t1, z1 = zip(*traj1.route)
    x2, y2, t2, z2 = zip(*traj2.route)
    P = zip(x1, y1)
    Q = zip(x2, y2)
    i = len(P)-1
    j = len(Q)-1
    ca = np.ones((len(P),len(Q)))
    ca = np.multiply(ca,-1)

    if ca[i,j] > -1:
        return ca[i,j]
    elif i == 0 and j == 0:
        ca[i,j] = euc_dist(P[0],Q[0])
    elif i > 0 and j == 0:
        ca[i,j] = max(_c(ca,i-
1,0,P,Q),euc_dist(P[i],Q[0]))
    elif i == 0 and j > 0:
        ca[i,j] = max(_c(ca,0,j-
1,P,Q),euc_dist(P[0],Q[j]))
    elif i > 0 and j > 0:
        ca[i,j] = min(_c(ca,i-1,j,P,Q),_c(ca,i-1,j-
1,P,Q),_c(ca,i,j-1,P,Q)) + euc_dist(P[i],Q[j])
    else:
        ca[i,j] = float("inf")

    distance = ca[i, j]

    print "DTW iterative results"
    print distance


    return ca[i,j]

Frechet Distance

def frechetDist(traj1,traj2):
    x1, y1, t1, z1 = zip(*traj1.route)
    x2, y2, t2, z2 = zip(*traj2.route)
```

```
P = zip(x1, y1)
Q = zip(x2, y2)
ca = np.ones((len(P),len(Q)))
ca = np.multiply(ca,-1)
return _c(ca,len(P)-1,len(Q)-1,P,Q)
```