Development of a Method to Merge Video Streams of Various Cameras with Rectilinear and Fisheye Lenses

ADP No. 150/20

Editors:	Cedric Derstroff	| 2937317
	Tobias Drebert	| 2336989
	Qiong Ge	| 2788113
	Li Liu	| 2972291
	Chenhao Tang	| 2570431
	Cheng-Ting Tsai	| 2968221
	Peng Yan	| 2734439
Supervisors:	Patrick Pintscher, M. Sc.	
	Cheng Wang, M. Sc.	

TECHNISCHE UNIVERSITÄT DARMSTADT

FZD FAHRZEUGTECHNIK TU DARMSTADT

Advanced Design Project Nr. 150/20 (6 CP)

für Li Liu
Cheng-Ting Tsai
Chenhao Tang
Qiong Ge
Peng Yan
Tobias Drebert
Cedric Derstroff

Voraussichtlicher Beginn: 16.11.2020
Bearbeitungsdauer: 13 Wochen

Thema: **Entwicklung einer Methode zum Zusammenfügen von Videostreams verschiedener Kameras mit geradlinigen und Fischaugen-Objektiven**

Topic: ***Development of a Method to Merge Video Streams of Various Cameras with Rectilinear and Fisheye Lenses***

Fachgebiet Fahrzeugtechnik

Prof. Dr. rer. nat. Hermann Winner

Otto-Berndt-Straße 2
64287 Darmstadt

Bearbeiter:
Patrick Pintscher M. Sc.
Tel. +49 6151 16 – 24231
Fax +49 6151 16 – 24205
patrick.pintscher@tu-darmstadt.de
www.fahrzeugtechnik-darmstadt.de

Datum
16.11.2020

Am Fachgebiet Fahrzeugtechnik der TU Darmstadt (FZD) wird in Zusammenarbeit mit der Industrie am Thema der Automatisierung und Teleoperation von Straßenbahnen geforscht. Für das Teleoperationssystem des Forschungsprojektes MAAS werden verschiedene Videostreams von Kameras mit geradlinigen und Fischaugen-Objektiven vom Fahrzeug zum sogenannten Telearbeitsplatz übertragen, von wo aus die Bahn ferngesteuert wird. Bislang werden diese Videostreams separat angezeigt.

Innerhalb dieser Gruppenarbeit soll eine Methode entwickelt sowie evaluiert werden, die die unterschiedlichen Videostreams zu einem einzigen zusammenführt und auf dem Bildschirm des Telearbeitsplatzes ausgibt.

Im Einzelnen sind folgende Arbeitsschritte auszuführen:

1. Einarbeitung in das Prinzip der Teleoperation, Objektivkorrektur in Videostreams sowie das perspektivische Zusammenführen dieser,
2. Präzisieren der Aufgabenstellung,
3. Analyse des Teleoperationssystems des Forschungsprojekts MAAS hinsichtlich der Videoübertragung,
4. Ermittlung der Anforderungen an die Methode zum Zusammenführen der Videostreams,

Dauer Zwischenkolloq (Vortrag / gesamt)
15 / 60 min

Dauer Abschlusskolloq (Vortrag / gesamt)
5 min pro Teammitglied / 60 min

TECHNISCHE UNIVERSITÄT DARMSTADT

5. Erarbeitung verschiedener Lösungskonzepte mit anschließender begründeter Auswahl für ein Konzept,
6. Umsetzung des Konzepts anhand zur Verfügung gestellter Videodaten aus dem Forschungsfahrzeug,
7. Evaluation des umgesetzten Konzepts

Schwerpunkte der Bewertung:

- Methodik des Vorgehens
- Vollständigkeit
- Nachvollziehbarkeit und Belastbarkeit der Argumentation
- Qualität folgender abzuliefernder Ergebnisse:
  - Schriftliche Ausarbeitung und Dokumentation
  - Quellcode zu eigener Software
  - Veränderungen/Ergänzungen an verwendeter Software von Dritten
- Abschlusskolloquium

Die Abgabe sämtlicher Messdaten und des Quellcodes wird vorausgesetzt. Die Arbeit bleibt Eigentum des Fachgebiets. Auf das Merkblatt des Fachgebiets wird hingewiesen.

Prof. Dr. rer. nat. Hermann Winner

Patrick Pintscher M. Sc.
(Betreuer)

Cheng Wang M. Sc.
(Co-Betreuer)

Cedric Derstroff
Matriculation No.: 2937317
Study program: Master Informatik

Tobias Drebert
Matriculation No.: 2336989
Study program: Master Informatik

Qiong Ge
Matriculation No.: 2788113
Study program: Master Maschinenbau

Li Liu
Matriculation No.: 2972291
Study program: Master Maschinenbau

Chenhao Tang
Matriculation No.: 2570431
Study program: Master Maschinenbau

Cheng-Ting Tsai
Matriculation No.: 2968221
Study program: Master Maschinenbau

Peng Yan
Matriculation No.: 2734439
Study program: Master Maschinenbau

ADP No. 150/20
Topic: Development of a Method to Merge Video Streams of Various Cameras with Rectilinear and Fisheye Lenses

Submitted: February 22, 2021

Technische Universität Darmstadt
Fachgebiet Fahrzeugtechnik
Prof. Dr. rer. nat. Hermann Winner
Otto-Berndt-Straße 2
64287 Darmstadt

Sworn Declaration

Erklärung zur Abschlussarbeit gemäß § 22 Abs. 7 APB TU Darmstadt

Hiermit versicheren wir, Cedric Derstroff, Tobias Drebert, Qiong Ge, Li Liu, Chenhao Tang, Cheng-Ting Tsai und Peng Yan, das vorliegende ADP gemäß § 22 Abs. 7 APB TU Darmstadt ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Uns ist bekannt, dass im Falle eines Plagiats (§38 Abs.2 APB) ein Täuschungsversuch vorliegt, der dazu führt, dass die Arbeit mit 5,0 bewertet und damit ein Prüfungsversuch verbraucht wird. Abschlussarbeiten dürfen nur einmal wiederholt werden.

English Translation for information purposes only:

Thesis Statement pursuant to § 22 paragraph 7 of APB TU Darmstadt

We herewith formally declare that we, Cedric Derstroff, Tobias Drebert, Qiong Ge, Li Liu, Chenhao Tang, Cheng-Ting Tsai and Peng Yan, have written the submitted thesis independently pursuant to § 22 paragraph 7 of APB TU Darmstadt. We did not use any outside support except for the quoted literature and other sources mentioned in the paper. We clearly marked and separately listed all of the literature and all of the other sources, which We employed when producing this academic work, either literally or in content. This thesis has not been handed in or published before in the same or similar form.

We are aware, that in case of an attempt at deception based on plagiarism (§38 Abs. 2 APB), the thesis would be graded with 5,0 and counted as one failed examination attempt. The thesis may only be repeated once.

Datum / Date: 22. Februar 2021

Cedric Derstroff	Tobias Drebert	Qiong Ge
Li Liu	Chenhao Tang	Cheng-Ting Tsai
Peng Yan		

Abstract

As a part of the MAAS project (Machbarkeitsstudie zur Automatisierung und zu Assistenzsystemen der Straßenbahn), this Advanced Design Project (ADP) mainly researches on the methods to stitch video streams of various cameras with rectilinear and fisheye lenses. The focus is to present the video data from the cameras of the tram to the operator in such a way, that the operator can easily understand the tram's environment and the tram's current state. With the stitched video stream, the operator can get an immersive operating experience, thereby improving the safety of teleoperation.

Because the video comes from three fisheye cameras and a pair of two stereo camera, a general consensus is that fisheye cameras will bring obvious image distortion while providing a larger field of view, which is not conducive to the operator's understand of the environment. Therefore, the fisheye camera needs to be corrected first, in this process the intrinsic camera parameters from the fisheye camera calibration are used.

In the next step, images should be stitched. Commonly used image stitching methods are mainly divided into two categories: pixel-based methods and feature-based methods, whereas deep learning approaches are currently on the rise as well. Selectively extracting feature points rather than all pixels in the overlapping regions, Feature-based methods are faster, more robust and more efficient. A mature feature-based method is global single transformation, it estimate the global homography relationship between two images, then images are mapped to a stitching surface and blended together. However, due to the large baseline between the MAAS tram cameras, the images of different cameras have large parallaxes, so this algorithm performance is poor.

Three local hybrid transformation methods were also investigated, including AANAP, NISwGSP and PtIS. AANAP and NISwGSP both focus only on local warp models when calculating similarity transformation, the difference is NISwGSP selects suitable scale and rotation parameters on the basis of AANAP to solve the problem of curved stitching results. PtIS combines the advantages of seam-driven methods, homography and content-preserving warping to handle parallaxes and avoid objectionable local distortions. These three methods are more robust than global single transformation, however, they can not get the perfect result under the condition of a huge baseline between the MAAS tram cameras. We have also investigated other image stitching methods, including using CNNs, 3D Modelling and Stitching and Vanishing Point Guided Natural Image Stitching. Due to lack of experimental conditions and insufficient time, we were actually not able to apply all of those. We have summarized their principles, nevertheless. If there is an opportunity to continue image stitching related work in the future, we can select the feasible ones among these methods to do related experiments.

Eventually we implemented fisheye camera synchronization and rectification functions in the ROS system. The architecture of ROS makes it more convenient to deploy this function on different platforms.

Contents

List of Abbreviations

a.k.a.	also known as
AANAP	Adaptive As-Natural-As-Possible
ADP	Advanced Design Project
APAP	As-Projective-As-Possible
AR	Augmented Reality
BRIEF	Binary Robust Independent Elementary Features
CF	Column Frequency
CNN	Convolutional Neural Network
DLT	Direct Linear Transformation
EU	European Union
FAST	Features from Accelerated Segment Test
FLANN	Fast Library for Approximate Nearest Neighbor
FoV	Field of View
fps	Frames per Second
FVID	Fusion Visual Information with Distortion Information
FVIND	Fusion Visual Information without Distortion Information
FZD	Fahrzeugtechnik Darmstadt
GIMP	GNU Image Manipulation Program
GPU	Graphics Processing Unit
GSM	Gaussian Scale Mixture
GSP	Global Similarity Prior
HoG	Histogram of oriented Gradients
HVS	Human Visual System
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
k-NN	k-Nearest Neighbors
LTE	Long Term Evolution
MAAS	Machbarkeitsstudie zur Automatisierung und zu Assistenzsystemen der Straßenbahn
MI	Mutual Information
MLDR	Minimum Line Distortion Rotation
MSE	Mean Squared Error
NISwGSP	Natural Image Stitching with the Global Similarity Prior
NMI	Normalized Mutual Information
NN	Neural Network

NSS	Natural Scene Statistics
NTP	Network Time Protocol
ORB	Oriented FAST and Rotated BRIEF
PSNR	Peak Signal to Noise Ration
PtIS	Parallax-tolerant Image Stitching
PTP	Precision Time Protocol
RAM	Random Access Memory
RANSAC	Random Sample Consensus
RF	Row Frequency
ROS	Robot Operating System
RPC	Remote Procedure Call
SF	Spatial Frequency
SIFT	Scale-Invariant Feature Transform
SLAM	Simultaneous Localization and Mapping
SSIM	Structural Similarity Index Measure
STD	Standard Deviation
SURF	Speeded Up Robust Features
TPS	Thin Plate Splines
TUDa	Technische Universität Darmstadt
VIF	Visual Information Fidelity
VIFF	Visual Information Fidelity for Fusion
VR	Virtual Reality
WLAN	Wireless Local Area Network

List of Figures

List of Tables

1 Introduction

In recent years, an increasing number of automotive companies, research institutes and universities around the world are developing autonomous driving technology. However, a fully driverless system in all situations is not yet in sight due to problems like safety concerns and government regulations[1]. Therefore, teleoperation, which describes the possibility of operating a machine remotely, can serve as a relevant solution for automated vehicles. Based on these situations, the Fahrzeugtechnik Darmstadt (FZD) of Technische Universität Darmstadt cooperates with the local transport company HEAG mobilo GmbH as well as HEAG Holding AG and Deutsche Telekom, and has promoted a project on the Feasibility Study for the Automation and Assistance Systems of Tramways (MAAS from the German phrase: Machbarkeitsstudie zur Automatisierung und zu Assistenzsystemen der Straßenbahn).[2]

1.1 Motivation

Since this Advanced Design Project (ADP) is part of the MAAS project, it is necessary to figure out what the goal of MAAS is. Aiming to scientifically identify the possibilities and limitations of automating tramways, MAAS investigates those technologies from other domains, such as automotive sector, which can be applied to trams. As completely automated driving is a long-term project, MAAS intends to build a prototype of a semi-automated, teleoperated driving system towards the end of the project, in order to research automation and teleoperation. Within this teleoperation system, the tram would be able run autonomously for most scenarios with assistance driving system, and only needs to be teleoperated for critical scenarios. During the whole driving process of tram, an operator workstation should connect to the tram via 4G/LTE[3](and maybe 5G in the future) from a distance. The basic components of teleoperation are shown in Fig. 1-1.

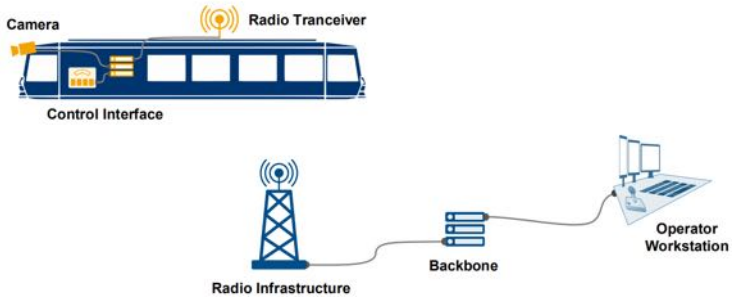


Figure 1-1: Basic Components of Teleoperation.[4]

Besides, a remote operator in the workstation has to observe the current state of tram's system as well as the objectives in the surrounding environments, such as vehicles, pedes-

[1] Georg, J. M. et al.: Teleoperated Driving (2018), p. 1.

[2] Pintscher, P.; Ruppert, T.: Feasibility Analysis for Automation (2021).

[3] AG, N.: NB3800 MediaRail (2019), p. 1.

[4] Pintscher, P.: Teleoperated Driving (2020), p. 13.

trians, traffic lights, etc. These information are firstly collected by the different cameras installed in the tram, then transmitted through the network communication, and finally presented to the operator. In consequence, the operator can take reaction according to various scenes as quickly as possible and send orders to the system in the tram.

From the structure data of the teleoperation tram, there are 3 fisheye cameras and 2 stereo cameras equipped in the tram. On the one hand, all three fisheye lenses can offer a wide-angle field of view so that show the environment around the tram. On the other hand, the stereo cameras can catch the distant vision and then provide depth information. Further information about the camera system in the tram is given in Subsec. 2.1.2.

Although the fisheye lens can provide the advantage of large field of view, the images from fisheye lens are severely distorted. In addition, all 5 frames from different cameras are presented separately. Thus, it is not a simple task for the operator to understanding the overall environmental scenarios around the tram well during the tramways operation. To meet the observation demands and shorten the reaction time of the operator, these video streams should be preprocessed to a more observable status. Therefore, this Advanced Design Project (ADP) mainly researches on the development of a method to stitch video streams of various cameras with rectilinear and fisheye lenses, in order to present the integrated reconstructed image on the monitor. Furthermore, the solution of presenting real driving scenarios has great significance for the realization of tram's teleoperated driving system and main project of MAAS.

1.2 Specification of the Task Description

The goal of this work is to obtain a panoramic view with three fisheye cameras and the stereo camera. This should support the operator at the operator workstation to control the tram. The different viewing angles of the cameras are intended to provide the operator with an all-round view. In the following, the task of the ADP is explained in detail.

With the help of the literature, a definition for teleoperation is elaborated and the way it works is explained. The challenges of teleoperation such as time delay and situation awareness are put in the context of the challenges of this project. Afterwards, some applications for teleoperation are mentioned. Furthermore, basic knowledge of lens correction and stitching in the video context will be researched.

Furthermore, the teleoperation system of the MAAS research project is analysed with regard to video transmission. The tram runs automatically in most situations and only needs to be teleoperated in certain situations. In this project, system latency is important. Therefore, the result of the algorithm for video equalisation and stitching must be available almost in real time to enable teleoperated intervention. It has to be weighed up whether the processing of the video data is done on the tram or on the operator workstation. To ensure that driving safety is not compromised, the quality (details preserved) of the video is important and freezing of the video must be avoided.

In this step, requirements for the method of merging the video streams should be determined which must be met or which are desirable. These requirements will later help in establishing advantages and disadvantages and in deciding on an algorithm.

Now different solution concepts are developed with the help of literature. The concepts are to be selected on the basis of the requirements justified developed. With simpler implementations, images can be used to check whether the result meets the requirements. If a method delivers good results but has a long run time, a more lightweight solution must be found, because otherwise remote control of the tram would be impossible.

The selected algorithms will be implemented in ROS so that the video streams can be processed. The component to be developed receives the camera images from the tram, and return the panoramic rectified image. If requirements cannot be met during implementation, other concepts can be tried.

The selected and implemented concept is to be evaluated finally. For this purpose, evaluation criteria are determined in advance and compromises as well as advantages and disadvantages of the solution are critically discussed. Among other things, it is important to check the degree of fulfillment of the functions required by the task.

1.3 Methodology

As a methodology for our approach to the project, we use an extended version of the waterfall model. This model is used because the phases can be clearly delineated and thus time planning is possible within the limited scope of the ADP. However, the extended version of the waterfall model is necessary, which allows a return to the previous phase, because there may be requirements that turn out to be unrealistic in the course of the project.

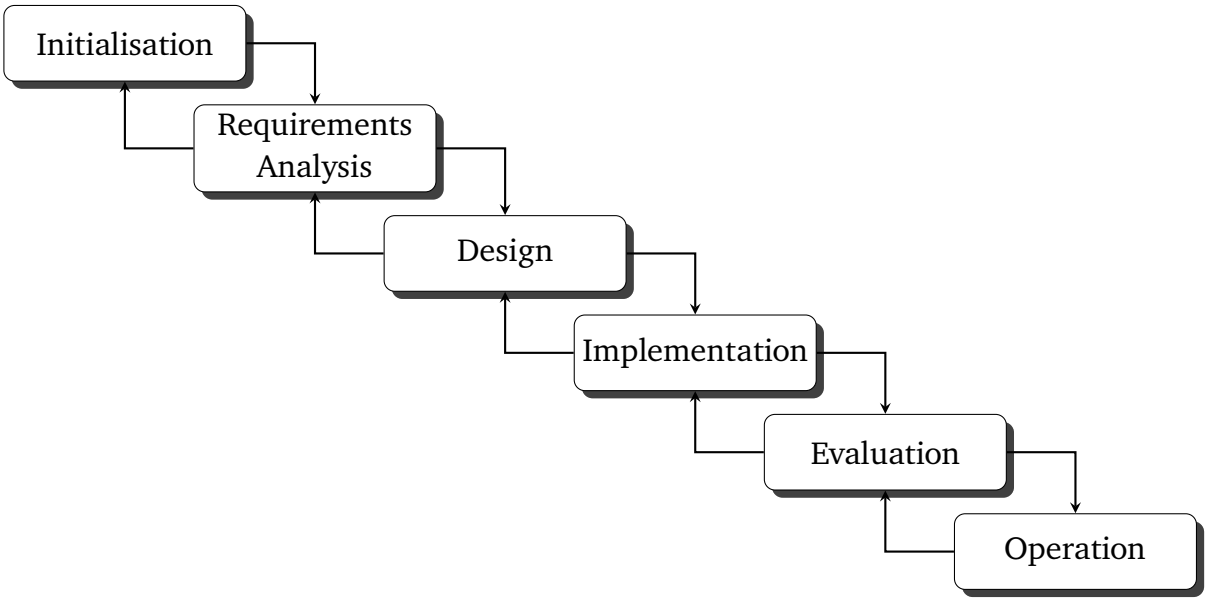


Figure 1-2: Methodology.

In the initialization phase, the task is specified and the subject matter is familiarized with the help of research. After a basic knowledge of the cameras and its field of application has been obtained, an analysis of the teleoperation system follows. With this analysis it is possible to develop requirements for the video stitching method of the camera streams. In the design, different algorithms are researched and tried out if necessary. The advantages and disadvantages in relation to the requirements are weighed to then decide on an approach. The selected approach with the goal of rectifying and stitching the camera images together will be implemented in ROS. The implementation is evaluated by analyzing the results and assessing the satisfaction of the requirements. If the results are convincing, the solution will be implemented in the MAAS system and will facilitate the operation of the tram at the operator workstation. At best, problems in the requirements, algorithms and evaluation are identified as early as possible so that they can be adapted and the goal of the application can be achieved.

2 Project Theories

Our ADP is part of the teleoperation system. The operator needs to obtain environmental information from the video. The focus is to present the video data from the cameras equipped in the tram to the operator in such a way, that the operator can easily understand the tram's environment and the trams current state. Fisheye camera can capture a wide field of view, but the significant barrel distortion, which results in noticeable bending of straightedges, and unnatural appearances of important features. This will bring extra labor to the operator to obtain video information, and the safety control of teleoperation will be affected. In addition, separated multiple cameras, including fisheye cameras and stereo cameras, will have overlapping images, this may cause misjudgment of the operator. Therefore, our main task is to develop a suitable algorithm to eliminate the distortion of the fisheye camera, and then stitch the images of multiple cameras to obtain a natural and smooth video stream, giving the operator an immersive experience. At the same time, the efficiency of the algorithm needs to be high enough to meet the requirements of low latency and ensure the safety of teleoperated driving.

2.1 MAAS Project

As is mentioned before, the final goal of MAAS is to build a prototype to research automated and teleoperated driving. Since self-driving is dangerous in critical scenarios, so teleoperation becomes a backup option in this situation. To analyze feasibility of tram's teleoperation, we need firstly know well about MAAS tram's basic conditions, especially the information of cameras, because the video information captured by the cameras is transmitted to the operator workstation. Then, the control orders are delivered back to the tram by the operator based on the video.

2.1.1 Basic Conditions of MAAS tram

This MAAS tram was built by Alstom, Vossloh Kiepe, and Bombardier in 2007[5] and is an eight-axle low-floor rail car. The track width is 1000 mm and the maximum speed is $70\,\mathrm{km\,h^{-1}}$, which is achieved on overland journeys.[6] Besides, the related sensors and operator workstation are the main components of this project. The sensors mainly include five cameras, two stereo cameras in the front of the tram, one fisheye camera in the middle of the tram, and the other two at each side of the tram. Fisheye cameras achieve extremely wide angles of view, but produce strong visual distortion at the same time. The stereo cameras simulate human binocular vision, and therefore provide the information of distance.

[5] GmbH, H. mobilo: Über 125 Jahre Nahverkehr in Darmstadt (2021).

[6] GmbH, H. mobilo: Die Straßenbahnen der HEAG mobilo (2021).

2.1.2 Cameras

In this section, more information of two types cameras are presented, namely the UI-5270CP-C-HQ Rev.2 (AB02036)[7] and the UI-5270FA-C-HQ (AB02048)[8]. The former is shown in Fig. 2-1b and used for the stereo camera setup and the latter, shown in Fig. 2-1a, for the three fisheye images. The details and specifications of the two different cameras from the manufacturer IDS are shown in Table 2-1. Although both camera types can output a resolution of 2056×1542 Pixel,the ROS camera node crops the images of the video stream to a resolution of 1625×1542 Pixel to cut of the part of the image which lacks image information.

(a) UI-5270FA-C-HQ (AB02048)[8] (b) UI-5270CP-C-HQ Rev.2 (AB02036)[7]

Figure 2-1: The two IDS cameras models used in the MAAS system without lenses. (a) is the fisheye camera and (b) is the stereo camera.

Table 2-1: Data sheet of IDS fisheye camera (UI-5270FA-C-HQ (AB02048))[7] and IDS stereo camera (UI-5270CP-C-HQ Rev.2 (AB02036))[8]

	Fisheye camera	Stereo camera
Name	UI-5270FA-C-HQ (AB02048)	UI-5270CP-C-HQ Rev.2 (AB02036)
Interface	Ethernet 1 Gbps	Ethernet 1 Gbps
Sensor type	CMOS	CMOS
Sensor manufacturer	Sony	Sony
Frame rate	36.0 fps	36.0 fps
Resolution (h×v)	2056×1542 Pixel	2056×1542 Pixel
Optical area	7.093 mm×5.320 mm	7.093 mm×5.320 mm
Optical class	1/1.8″	1/1.8″
Resolution	3.17 MPixel	3.17 MPixel
Pixel size	3.45 µm	3.45 µm
IP code	IP65/67	IP30

Mounted on the stereo camera, there is the #68-670 lens by Edmund Optics Inc. with

[7] IDS: UI-5270CP Rev. 2 (2021).

[8] IDS: UI-5270FA (2021).

a focal length of $f = 5\,\text{mm}$ and an $f2.8$ aperture[9]. For the fisheye camera, the fisheye lens Lensagon BF10M19828S118 by Lensation GmbH is used. It has a focal length of $f = 1.89\,\text{mm}$, an $f2.8$ aperture and yields a Field of View (FoV) of $180°$[10].

The three fisheye cameras are mounted on top of the tram. Like shown in Fig. 2-2, two cameras are positioned at either side and viewing the sides, the last one is mounted at the front facing forward. The stereo camera setup is placed in the cockpit under the line information sign and right behind the wind shield.

Figure 2-2: Camera setup on the MAAS tram[11]. The fisheye cameras are highlighted in red and the stereo cameras in yellow.

Figure 2-3 shows the view of the three aforedescribed fisheye lenses in an example scene in the top row ((a)-(c)). The corresponding output of the stereo images is shown in the bottom row ((d) and (e)). It is evident that the stereo images look natural, i.e. what humans are used to, and the fisheye images look unnatural and distorted. First and foremost to mention is the fact that straight lines in the scene are curved in the image and lines in the scene parallel to the image plane are not parallel in the images. For the stereo images on the other hand, this mostly holds true.

[9] Edmund Optics GmbH: 5mm Brennweite, Weitwinkelobjektiv mit geringer Verzeichnung (2021).

[10] Lensation GmbH: Lensagon BF10M19828S118 – Lensation GmbH (2021).

[11] Pintscher, P.: Teleoperated Driving (2020), p. 5.

(a) fisheye-left

(b) fisheye-middle

(c) fisheye-right

(d) stereo-left

(e) stereo-right

Figure 2-3: Example images of the MAAS camera setup. The three fisheye images are displayed in the top row and the stereo images in the bottom row.

2.2 Teleoperation

Teleoperation describes the remote control of a robot or vehicle. It has been a research topic for several decades[12] and it is used in various application areas. In leisure time, it is often used as a sport for fun[13]. In industry, it is mainly used in areas that are difficult for humans to reach or too dangerous[14,12]. Current applications of teleoperation include but are not limited to space exploration, dangerous military applications like land-mine field clearing vehicles and unmanned air vehicles (commonly known as drones) and underwater applications[15]. Furthermore, teleoperation is an important step from manual to automated driving[16,13].

The most important aspects of teleoperation which needs to be handled with care are *Latency/Delay*, *Data rate*, *Situation awareness/telepresence*, *Safety* and *Security*[13,16]. Each of these will be further discussed in its corresponding part in the following.

2.2.1 Latency and Time Delay

Latency is one of the most important factors for the teleoperation[16], it will not only affect the operating experience of the operator, but more importantly, due to the latency, the

[12] Georg, J. M. et al.: Teleoperated Driving (2018).

[13] Pintscher, P.: Teleoperated Driving (2020).

[14] Niemeyer, G. et al.: Telerobotics (2016).

[15] Lichiardopol, S.: A Survey on Teleoperation (2007).

[16] Neumeier, S. et al.: Teleoperation: The Holy Grail to Solve Problems of Automated Driving? (2019).

effective distance generated during the execution of teleoperation will seriously affect its safety. For teleoperation, the latency mainly includes perception latency and reaction latency. Perception latency means that the video frame of the camera obtained on the screen seen by the operator is not taken at the current moment, and the status of the tram may have changed at this time. Because of the physical limitation of data transmission speed, the control instructions issued by the operator from the working station cannot be executed immediately. When the operator's instructions actually take effect, the status of the tram may also change, this is reaction latency. For our project, we focus on the reasons for the formation of perception latency[17].

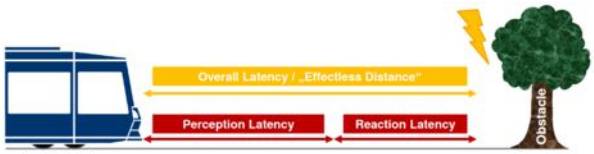


Figure 2-4: Effectless distance caused by latency.[17]

Through the analysis of tram video transmission system, perception latency includes the following five sources.

Camera Frame Rate

The camera cannot record image information continuously, frame rate is the amount of individual video frames that camera captures per second. Therefore, the changes in the current tram status may not be recorded in time, and there will be latency.

Video Rectification and Stitching

Video rectification and stitching are the main tasks that our project needs to complete, and they are also parts that need to be optimized using good algorithms. The rectification and image stitching of the fisheye lens occupy huge computing resources while also requiring a certain amount of computing time.

Video Compression

In order to use limited bandwidth to ensure the smooth transmission of video under complex network conditions, video information must be compressed. There are many video compression techniques, including lossy and lossless compression. Using lossless compression, the video can be compressed without losing quality[17]. This compression process is also reversible, we can reverse the process to get the original video data. However, the compression ratio is usually not better than 2:1, although some approaches achieve 3:1 or 4:1[18]. In contrast, lossy compression achieves large compression ratio while reducing the quality of the video. The compression ratio of lossy compression can be as high as

[17] Pintscher, P.: Teleoperated Driving (2020), p. 10.

[18] Mittal, S.; Vetter, J. S.: A Survey Of Architectural Approaches for Data Compression (2016).

200:1[19]. Although there are many excellent compression algorithms, a certain amount of latency will also occur.

Data Transfer

When a packet travels from one node (host or router) to the subsequent node (host or router) in a computer network, the packet experiences several different types of latency at each node along the path, including Nodal Processing Latency, Queuing Latency, Transmission Latency, Propagation Latency. These four latency values add up to the Total Nodal Latency[20].

Display Refresh Rate

The refresh rate is the number of times the monitor updates with new images each second. The desktop monitor standard is a 60 Hz refresh rate, but in recent years more specialized, high-performing monitors also support 120 Hz, 144 Hz and even 240 Hz refresh rates. The display refresh rate of the monitor used in the MAAS project is 120 Hz[21].

2.2.2 Data Rate

The data rate strongly correlates with other main aspects of teleoperation. On the one hand, like stated in the previous part, the data rate is one factor of the latency. On the other hand, the data rate limits the information about the environment that can be transmitted at a time and therefore has an indirect influence on the telepresence which will be described in the next part.

The visual information is the most important for a human operator. Unfortunately, images are especially expensive regarding data rate. Hence, the data rate strongly regulates the resolution and the number of cameras that can be used for the teleoperation setup.

2.2.3 Situation Awareness and Telepresence

Telepresence basically stands for the operator feeling present at the operator site or remote world, in our case at the tram in the traffic. He has to create a representation of the vehicles environment in his mind. Although some kind of telepresence can already be achieved by a simple camera-monitor setup, the operator workstation usually features a way more sophisticated setup including Virtual Reality (VR) or Augmented Reality (AR) setup, force feedback and auditory stimulation.[22] Based on the data privacy regulations in Germany and the European Union (EU), microphones cannot be included in the sensor setup of the MAAS-tram[23]. 90 % of the information a human perceives is through vision[22], thus, cameras play an important role in telepresence. Telepresence also includes situation

[19] Henry, M.: Video Compression Explained (2010).

[20] Ross, K. W.; Kurose, J. F.: Delay and Loss in Packet-Switched Networks (2000).

[21] Samsung Electronics GmbH: datenblatt LC49RG94SSUXZG (2019).

[22] Lichiardopol, S.: A Survey on Teleoperation (2007).

[23] This was given in a meeting with Mr. Pintscher and Mr. Wang of FZD.

awareness. One widely used definition by Mica Endsley "Situation Awareness is the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future."[24]. This definition is widely used and basically means being fully aware of the situation the teleoperated machine (vehicle or robot). It is even more important than in a direct setup since the operator can not directly sense the consequences of his actions and is limited to the noisy information channels available and cannot sense the environment and situation on all his senses.

Further, telepresence and situation awareness is also strongly correlated with the operator load. When the operator cannot easily immerse into the situation, the operator needs more energy to concentrate and focus on the task which leads to operator fatigue and a higher stress level. [25]

2.2.4 Safety

Apparently, teleoperation always maximally increases the safety of the operator. A setup that involves vulnerable creatures, it is a high priority to avoid any harm to those. Especially, in teleoperating vehicles in traffic situation this is the case. Situation awareness and telepresence are basic requirements to ensure everybody's safety. In addition, it is possible to increase the safety by supporting the operator with assist functionality like emergency break systems.

2.2.5 Security

Security is an important factor in every technical application and thus, it also holds true for teleoperation systems. The developers of the teleoperation system have to ensure that nobody can hack into the system from afar or aboard the system. In the beginning of this subsection, we already gave some example application of teleoperation. From these, it is evident that malicious take over in most cases not only causes the loss of a high amount of money but also affects the safety humans or animals. For the MAAS project a foreign takeover would not only mean the loss of an expensive machine but also a severe threat to all the passengers inside the tram. There are several methods to provide security. The hardware should be physically protected, so nobody can alter the system. Further, the communication also must be secure. The way to enforce the highest security is to use a wired direct data transfer via cable. This also gives the benefit of of a low latency. In many applications, however, this is not applicable and one has to use other wired or wireless technologies like WLAN, cellular radio and the network of "Internet Service Providers". To protect these communication methods from hostile acquisition, it s usually encrypted and can be further secured by authentication methods[26].

[24] Endsley, M. R.: Situation awareness global assessment technique (SAGAT) (1988).

[25] Georg, J. M. et al.: Teleoperated Driving (2018).

[26] Pintscher, P.: Teleoperated Driving (2020), p. 23.

2.3 ROS System

2.3.1 ROS Introduction

The Robot Operating System (ROS) is an open-source, meta-operating system specially designed for robot development. It provides a series of functionalities, including hardware abstraction, low-level device control, message-passing between processes, and package management. It also provides tools and libraries for managing code across multiple computers.[27]

The ROS processing architecture uses the ROS communication infrastructure to implement a peer-to-peer loosely coupled network. ROS implements several different styles of communication, including synchronous Remote Procedure Call (RPC)-style communication over services, asynchronous streaming of data over topics, and storage of data on a Parameter Server.[27]

ROS supports the reuse of code in research and development of robotics. ROS is a distributed framework of processes (aka Nodes) that enables subsystems to be individually designed and loosely coupled at run time. These processes can be grouped into Packages and Stacks to be easily shared and distributed. ROS also supports a federated system of code Repositories that enable collaboration to be distributed as well. The file system level to the community level of ROS system, enables independent decisions about development and implementation, and all can be integrate with ROS infrastructure tools.[27]

In order to support the main goal of sharing and collaboration, there are several other goals of the ROS framework:

1. Thin: ROS is designed to be as thin as possible, main framework and architecture will not be changed, so that code written for ROS can be used with other robot software frameworks. A corollary to this is that ROS is easy to integrate with other robot software frameworks: ROS has already been integrated with OpenRAVE[28], Orocos[29], and Player[30].
2. ROS agnostic libraries: write ROS-agnostic libraries with clean functional interfaces is the preferred in development.
3. Language independence: in ROS framework we can use any modern programming language. ROS has already been implemented in Python, C++, and LISP, and experimental libraries in Java and Lua are also available.
4. Easy testing: ROS use a builtin test framework called rostest that makes it easy to bring up and tear down test fixtures.
5. Scaling: ROS is appropriate for large projects.

[27] Open Source Robotics Foundation, Inc.: ROS/Introduction (2018).

[28] http://openrave.org.

[29] https://orocos.org.

[30] http://playerstage.sourceforge.net.

2.3.2 ROS Computation Graph

The Computation Graph is the peer-to-peer network of ROS processes. Nodes, Master, Parameter Server, messages, services, topics, and bags are the basic concepts of ROS Computation Graph, they provide data to the Graph in different ways.[31]

ROS Nodes

Nodes perform the computation processes. ROS is designed to be modular at a fine-grained scale and a robot control system usually comprises many nodes. For example, one node controls a camera driver, one node controls the image processing module, one node performs image visualization, and so on. A ROS node can be written with in different programming language, such as C++and Python utilizing roscpp and rospy respectively.[31]

ROS Master

The ROS Master makes name registration and lookup to the rest of the Computation Graph possible. Nodes would be able to find each other, exchange messages, or invoke services With the Master.[31]

ROS Parameter Server

As a part of the ROS Master currently, the Parameter Server allows data to be stored by key in a central location.[31]

ROS Messages

Nodes use messages to communicate with each other. A message comprise typed fields likes a data structure. Standard primitive types (integer, floating point, boolean, etc.) are supported. Messages can include arbitrarily nested structures and arrays like C structs.[31]

ROS Topics

Messages are passed with a transport system with publish/subscribe semantics. A node sends out a message by publishing it to a given topic, which is a name that is used to identify the content of the message. A node that is interested in a certain kind of data will subscribe to the appropriate topic. There may be multiple concurrent publishers and subscribers for a single topic, and a single node publishes and/or subscribes to multiple topics is also possible. Publishers and subscribers are not aware of each others' existence, thus the production of information is decoupled from its consumption. Logically, one can think of a topic as a message bus. Each bus has a name, anyone if they are the right type can connect to the bus to send or receive messages.[31]

[31] Open Source Robotics Foundation, Inc.: ROS/Concepts (2014).

ROS Services

The publish/subscribe model is a very flexible many-to-many communication paradigm, one-way transport is more suitable for request/reply interactions, which are often required in a distributed system. ROS services achieves Request/reply interactions. They are defined by a pair of message structures: one for the request and one for the reply. A node offers a service under a name and a client uses the service by sending the request message and waiting for the reply. This interaction is like a remote procedure call.[32]

ROS Bags

Bags are an important mechanism for storing and playing back data, such as sensor data, that can be difficult to collect but is necessary for developing and testing algorithms.[32]

2.4 Camera Model and Distortion Correction

The following part of this work deals with the basics of camera projection and of distortion correction (a.k.a. lens correction or image rectification).

2.4.1 Coordinate System of Fisheye Lens

To describe the position of image and object point and simplify the process of calculation, four coordinate systems are introduced here, world coordinate system, camera coordinate system, image coordinate system, and pixel coordinate system. Transformation from one coordinate system to another can be realized by multiplying corresponding matrix. Through calibration of cameras, these matrices are divided into intrinsic- and extrinsic matrices, which are seen as the parameters of the relationship of projection. World coordinate system $O_{fl-world}$ describe the real position of object point in the world. In 3D world, the point of objective point shows as $(X_{fl-world}, Y_{fl-world}, Z_{fl-world})$. To transform into camera coordinate system $O_{fl-camera}$, which shows the position of objective point referring to camera, the rotation matrix $\mathbf{R}$ and translation matrix $\mathbf{T}$ of camera should be taken into consideration. The multiplied matrix is known as extrinsic matrix. After projection into image plane, the position of image point is defined with image coordinate system $O_{fl-image}$ with two dimension. To describe the position in picture with the help of pixel, image coordinate system $O_{fl-image}$ is transformed into pixel coordinate system $O_{fl-pixel}(u, v)$, which is important for the image rectification calibration process. And the matrix from camera coordinate system to pixel coordinate system is known as intrinsic matrix. With those considerations, the matrix system is rewritten and passes to a more general case, which can be shown as Eq. (2-1):

$$s \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{pmatrix} \begin{pmatrix} X_{fl-world} \\ Y_{fl-world} \\ Z_{fl-world} \\ 1 \end{pmatrix} \tag{2-1}$$

[32] Open Source Robotics Foundation, Inc.: ROS/Concepts (2014).

s: Scale coefficient
f_x: Width direction focal length of the camera
f_y: Height direction focal length of the camera
c_x: The abscissa value of the optical center of the camera
c_y: The ordinate value of the optical center position of the camera

2.4.2 Projection Model

When the lens is produced, the fisheye lens is designed according to a certain pattern. The theoretical model of the perspective projection of a pinhole camera can be described by the following formula[33]:
Perspective projection:

$$r = f\tan(\theta) \tag{2-2}$$

$\theta$is the angle of incidence in the centre of lens. f is the distance between image plane and the centre of fisheye lens, which is known as the focal length. With different projection models, the distance between the image point and the image principle point r is different. In fisheye lens, there are more non-perspective projection models used.
Stereographic projection:

$$r = 2f\tan(\frac{\theta}{2}) \tag{2-3}$$

Equidistance projection:

$$r = f\cdot\theta \tag{2-4}$$

Equisolid angle projection:

$$r = 2f\sin(\frac{\theta}{2}) \tag{2-5}$$

Orthogonal projection:

$$r = f\sin(\theta) \tag{2-6}$$

Figure 2-5 shows the projections of different lenses, p, p_1 , p_2, p_3 and p_4 are perspective projection, stereographic projection, equidistance projection, equisolid angle projection and orthogonal projection. The corresponding distances are r, r_1 , r_2, r_3, r_4.

[33] Hou, W. et al.: Digital deformation model for fisheye image. (2012), p. 3.

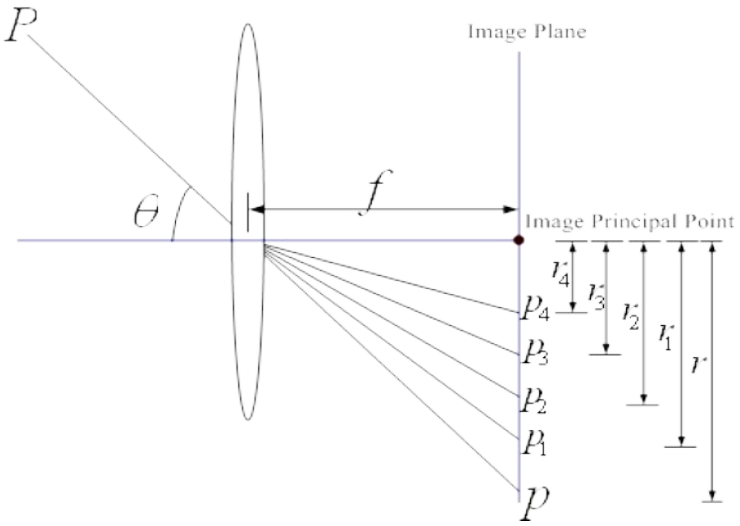


Figure 2-5: The projections of different models.[34]

However, the real lens don't exactly follow the designed projection model. To simulate different projection models and make calibration more efficient, polynomial function based on Taylor series can be used here:

$$r(\theta) = k_1\theta + k_2\theta^3 + k_3\theta^5 + k_4\theta^7 + k_5\theta^9 + \cdots \tag{2-7}$$

To get good approximation of different projection curves, the power of θ is set to ninth and five parameters of k, which all mean distortion coefficients, should be fixed here.

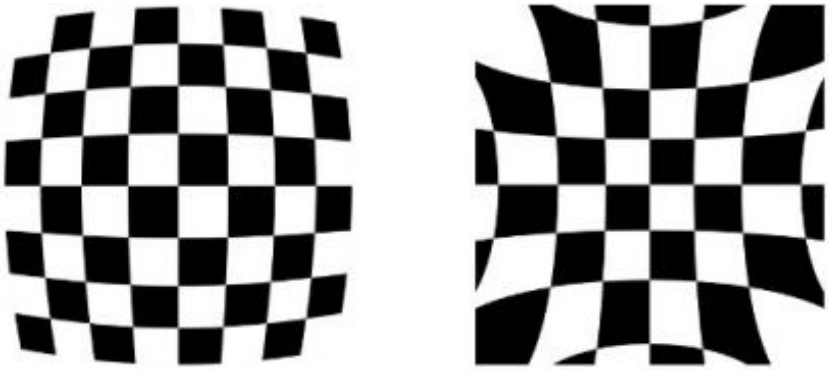
2.4.3 Distortion Types and Mathematical Models

Fisheye lens has the prominent advantage of large field of view, which is suitable for the overall environmental monitoring during the teleoperated driving of tram. As a result of fisheye lens imaging characteristics, the shooting images will introduce a significant "projection distortion", intuitive feeling is linear object in the image will become a curve. This large distortion has a great influence on the image matching and stitching between different cameras.[35] Generally, fisheye cameras have three types of distortion, namely radial distortion, tangential distortion and thin prism distortion.

Radial distortion is mainly caused by the lens itself, as the magnification in the center area of the optical axis is not consistent with the edge area. If the magnification of the central area of the optical axis is much greater than the edge area, rays far from the center of the lens will bend more severely than rays close to the center. This phenomenon is called barrel distortion, as shown in Fig. 2-6(a), and often occurs in wide-angle lenses and fisheye lenses. On the contrary, if the magnification of the edge area is greater than the central area of the optical axis, pincushion distortion will occur, as shown in Fig. 2-6(b), which is common in telephoto lenses.

[34] Hou, W. et al.: Digital deformation model for fisheye image. (2012), p. 4.

[35] Liu, Y. et al.: Fisheye image distortion correction. (2020).



(a) Barrel distortion (b) Pincusion distortion

Figure 2-6: Checkerboards with two types of radial distortions.[36]

Tangential distortion is mainly caused by the installation of the lens. When the lens is not completely parallel to the image plane, tangential distortion will occur.[36]

The distortion of thin prisms is caused by lens design, processing and installation errors, and it can be ignored in general.

According to the explanation in OpenCV documentation[36], radial distortion mathematical model can be described as follows:

$$x' = x\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right) \tag{2-8a}$$

$$y' = y\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right) \tag{2-8b}$$

Tangential distortion mathematical model:

$$x' = x + \left[2p_1 y + p_2\left(r^2 + 2x^2\right)\right] \tag{2-9a}$$

$$y' = y + \left[2p_2 x + p_1\left(r^2 + 2y^2\right)\right] \tag{2-9b}$$

Total distortion mathematical model, which consists of radial and tangential distortion, is consequently displayed like this:

$$x' = x\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right) + 2p_1 xy + p_2\left(r^2 + 2x^2\right) \tag{2-10a}$$

$$y' = y\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right) + 2p_2 xy + p_1\left(r^2 + 2y^2\right) \tag{2-10b}$$

Besides, we can obtain the pixel coordinates:

$$u' = x f_x + c_x \tag{2-11a}$$

$$v' = y f_y + c_y \tag{2-11b}$$

[36] OpenCV: Camera Calibration and 3D Reconstruction (2021).

r: Distortion radius
(x,y): The original position of the distortion point in the image plane
(x',y'): The new position of the distortion point after correction
k_1,k_2,k_3: Radial distortion coefficient
p_1,p_2: Tangential distortion coefficient
(u',v'): Pixel coordinates of distortion point after correction

2.4.4 Distortion Correction Process

Although the images captured by the fisheye lens have large distortion, the one-to-one correspondence between the object space and the image space is still strictly maintained, so that distortion can be eliminated as much as possible through subsequent correction, and preparations are made for subsequent image processing algorithms. In order to achieve video streams stitching, we must first transfer the distorted image to undistorted one, as is shown in Fig. 2-7.

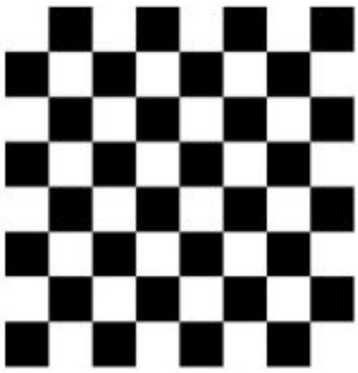


Figure 2-7: Undistorted checkerboard.[37]

Generally, we need to obtain camera intrinsic matrix and distortion coefficients by calibration before we correct the fisheye distortion. Since the intrinsic parameters are characters of the camera itself, each camera has different intrinsic parameters. So we have to calibrate every fisheye lens separately. In OpenCV, the calibration algorithm consists of four main steps, which are listed in Fig. 2-8.

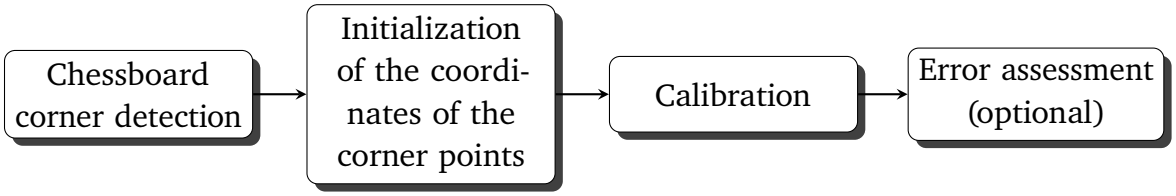


Figure 2-8: Fisheye calibration process.[38]

[37] OpenCV: Camera Calibration and 3D Reconstruction (2021).

[38] Kannala, J.; Brandt, S. S.: A generic camera model and calibration method (2006)

Through intrinsic calibration, we can obtain intrinsic camera parameters[39], which are listed in Table 2-2.

Table 2-2: Camera parameters explanation

Parameters	Explanation	Form
K	Camera intrinsic matrix.	$\mathbf{K} = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix}$
D	Vector of distortion coefficients.	$\mathbf{D} = \begin{pmatrix} k_1 & k_2 & k_3 & k_4 \end{pmatrix}$
R	Rectification transformation in the object space.	$\mathbf{R} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$
P	New projection matrix.	$\mathbf{P} = \begin{pmatrix} f_x & 0 & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$

According to the known camera intrinsic matrix and distortion coefficients, we can then determine the mapping relationship between the distorted image and the target image. The main process functions as shown in Fig. 2-9. Under this condition we usually use the reverse mapping method to complete the undistortion process. This method is to start from each point on the original distorted image, according to the deformation relationship of the two images, and then use the point information to assign values to the pixels of the target image.

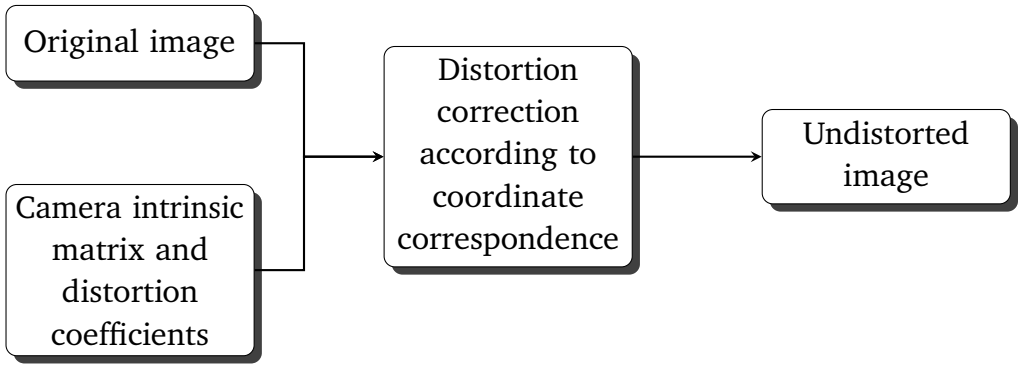


Figure 2-9: The lens correction process using the intrinsic camera parameters.

2.5 Image and Video Stitching Theories

Image stitching (a.k.a. mosaicing or panorama stitching) describes the process of combining several images to a panoramic image. Owing to the direct relationship between image and video, video stitching is in many ways an extension or generalization of multi-image stitching. Besides, the large degrees of independent motion, camera zoom, and the desire to visualize dynamic events impose additional challenges.

[39] OpenCV: Fisheye camera model (2021).

Generally, video stitching algorithms involve three steps. Firstly, a stitching template is constructed by stitching the selected frames of original videos with image stitching algorithms. Secondly, a single wide-angle video is generated by stitching subsequence frames according to the template. Finally, foreground detection should be employed to solve the potential blurring and ghosting in the stitched video, i.e., the stitching template is updated when an object moves across the overlapping regions among images[40].

Image stitching is one of the most important and most widely used topics in computer vision and graphics. In last decades, stitching algorithms have been applied in many fields, such as image processing, computer vision, and multimedia. Many well-known applications, such as Adobe Photoshop, AutoStitch[41], PTGui[42], and Image Composite Editor (ICE)[43] effectively stitch multiple overlapping images to generate a wide-angle view. Image mosaicing algorithms traditionally follow a structural alignment approach, involving warping and stitching.These are mainly divided into two categories: pixel-based methods, and feature-based methods[44a]. The classification of image stitching methods is displayed in Fig. 2-10.

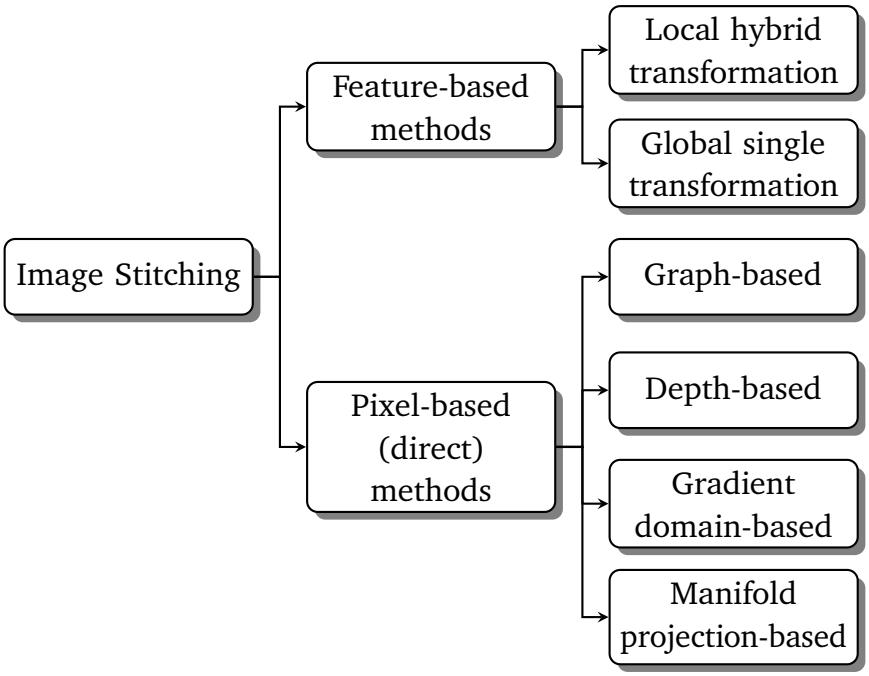


Figure 2-10: Classification of image stitching.[44b]

[40] He, B.; Yu, S.: Parallax-Robust Surveillance Video Stitching (2015).

[41] http://matthewalunbrown.com/autostitch/autostitch.html.

[42] https://www.ptgui.com.

[43] https://www.microsoft.com/research/project/image-composite-editor.

[44] Lyu, W. et al.: A survey on image and video stitching (2019), a: p. 1; b: p. 5.

2.5.1 Pixel-based Stitching

Owing to the advantage of image information, pixel-based methods (a.k.a. direct methods) register multiple images by directly minimizing pixel-to-pixel dissimilarities[45]. These methods estimate a global transformation model to deform and align images, such as 3D rotation matrix of camera.

For instance, because the gradient information is so sensitive to high-level features in the image, such as lines, contours, and edges, that some image stitching algorithms adopt a coarse-to-fine strategy used until now to optimize the approximate alignment in the gradient domain[46]. Furthermore, the depth and color of pixels are utilized[47], and some other works adopt the optical flow method and formulate the stitching as a manifold projection, whereby images are divided into many strips according to the calculated optical flow vector[48]. Moreover, a graph structure has also been adopted to assist the stitching[49].

However, these methods effectively register the image according to image information, but require complex pre-processing and large calculations, such as intrinsic and extrinsic camera calibration. Particularly, these pixel-based stitching methods can only address images with simple scenes contained in the same plane. Obviously, these are not suitable for the actual situation of our project, because these video streams are supposed to help the operator understanding the overall environment around the tram. In other words, these scenarios captured by the cameras undoubtedly include different objects in different planes. In consequence, we will not adopt pixel-based stitching and there is no need to introduce more details about these methods.

2.5.2 Feature-based Stitching

Different from direct methods, feature-based stitching employs the sparse feature descriptors and feature matching is implemented sequentially. By selectively extracting feature points rather than all pixels in the overlapping regions, feature-based methods are definitely faster, more robust and efficient than pixel-based stitching. Next we will introduce different methods used for the detection of feature points as well as feature matching.

Edge/Corner Detectors

Edges and Corners[50a] are the most important features that could be extracted from photos. As mentioned before the gradient of photo will be calculated:

$$\nabla g(x,y) = \left(\frac{\partial g(x,y)}{\partial x}, \frac{\partial g(x,y)}{\partial y} \right) \tag{2-12}$$

[45] Jiaya Jia; Chi-Keung Tang: Eliminating structure and intensity misalignment (2005).

[46] Levin, A. et al.: Seamless Image Stitching in the Gradient Domain (2004).

[47] Zhi, Q.; Cooperstock, J. R.: Toward dynamic image mosaic generation (2012).

[48] Peleg, S. et al.: Mosaicing on adaptive manifolds (2000).

[49] Uyttendaele, M., Eden, A.,; Skeliski, R.: Eliminating ghosting and exposure artifacts (2001).

First, we pull out useful information from the original photo (Fig. 2-11 left) by this formula. The gradient is partially differentiated with respect to two directions x and y. At this point we use one filter, called Sobel operator, to approximate the calculation of gradient. Also it could "smooth" photos so that influences from image noises, which cause the error of non-existing edges, decrease. As next step the edited photos of two directions should be added, and this photo shows only gradients which need to be judged where the contours locate. At last step, though defined threshold different gradients will be classified: if the amount surpass the threshold, this point is read as edge, and vice versa as nothing. The above process is so-called Canny edge detector(Fig. 2-11a). It provides one simple way for edge detection and makes itself as representative in this area.

Figure 2-11: Comparison of different detectors.
left: input image, a: with Canny edge detector, b: with Harris corner detector[50b]

Another detector called Harris corner detector (Fig. 2-11b). The gradients of original photos are also calculated. The difference from Canny detector is that the gradient intensities in all directions are taken into account. If the accumulation of these amounts surpass a certain value, it will be judged as a corner.
Although Canny or Harris detectors are easily applied for the tasks, it yields one disadvantage while the interested objects should be found. The contours of unimportant objects will be detected and cause unnecessary time waste for computing. Therefore, more efficient method for finding feature points should be proposed for use.

SIFT

The Scale-Invariant Feature Transform (SIFT)[51] descriptor was computed from the image intensities around interesting locations in the image domain which can be referred to as interest points, alternatively key points. Different from edges and corners, this method is aimed at detecting regions in a digital image that differ in properties, such as brightness or color, compared to surrounding regions. A Gaussian pyramid is constructed from the input image by repeated smoothing and subsampling, and a difference-of-Gaussians pyramid is computed from the differences between the adjacent levels in the Gaussian pyramid. Then, interest points are obtained from the points at which the difference-of-Gaussians values assume extrema with respect to both the spatial coordinates in the image domain and the

[50] Winner, H.: Handbuch Fahrassistenzsysteme (2015), a: p. 374; b: p. 375.

[51] Lindeberg, T.: Scale Invariant Feature Transform (2012).

scale level in the pyramid. In other words, SIFT chooses filters (Gaussian kernels) with different scales to approximate gradients (Gaussian convolutions), as shown in Fig. 2-12.

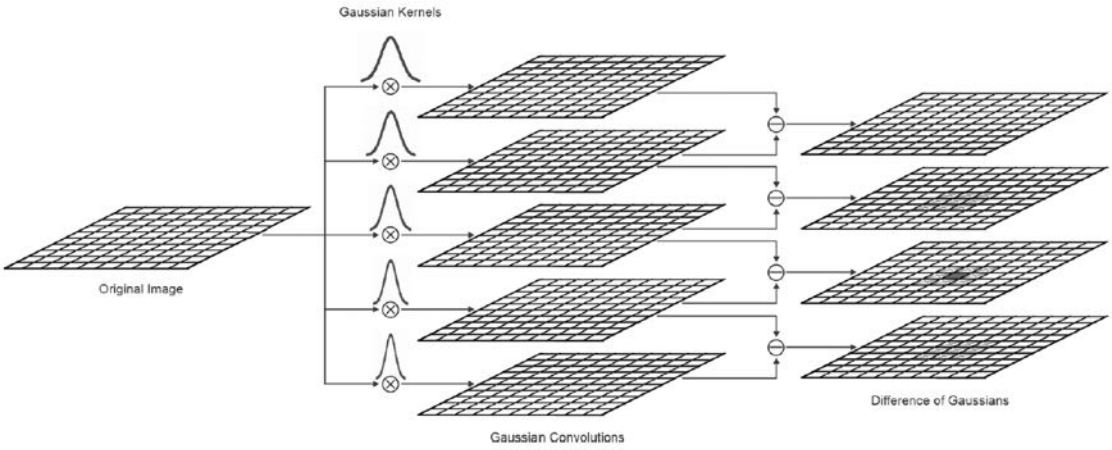


Figure 2-12: Overview of SIFT.[52]

This method for detecting interest points in the SIFT operator can be seen as a variation of a scale-adaptive blob detection, where blobs with associated scale levels are detected from scale-space extrema of the scale-normalized Laplacian. The scale-normalized Laplacian is normalized with respect to the scale level in scale-space and is defined as Eq. (2-13).

$$\nabla^2_{norm} L(x,y;s) = s(L_{xx} + L_{yy}) = s(\frac{\partial^2 L}{\partial x^2} + \frac{\partial^2 L}{\partial y^2}) = s\nabla^2(G(x,y;s) * f(x,y)) \tag{2-13}$$

In Eq. (2-13), $L(x,y;s)$ means smoothed image values and its scale-normalized Laplacian is computed from the image input $f(x,y)$ by convolution with Gaussian kernels, which can be described as Eq. (2-14):

$$G(x,y;s) = \frac{1}{2\pi s} e^{-(x^2+y^2)/(2s)} \tag{2-14}$$

where, different widths of the Gaussian kernels $s = \sigma^2$ and σ denotes the standard deviation. Then, the difference-of-Gaussians operator constitutes an approximation of the Laplacian operator as shown in Eq. (2-15).

$$DOG(x,y;s) = L(x,y;s+\Delta s) - L(x,y;s) \approx \frac{\Delta s}{2}\nabla^2 L(x,y;s) \tag{2-15}$$

By a self-similar distribution of scale levels $\sigma_(i+1) = k\sigma_i$, an approximation of the scale-normalized Laplacian was constituted with Eq. (2-16)[53a] :

$$\Delta s\nabla^2 L = (k^2-1)t\nabla^2 L = (k^2-1)\nabla^2_{norm} L \tag{2-16}$$

[52] Kaehler, A.; Bradski, G.: Learning OpenCV 3 (2016), p. 541.

therefore applying:

$$DOG(x,y;s) \approx \frac{(k^2-1)}{2}\nabla^2_{norm}L(x,y;s) \tag{2-17}$$

It can be shown that this method for detecting interest points leads to scale-invariance because the interest points are preserved under scaling transformations and the selected scale levels are transformed in accordance with the amount of scaling.

To minimize the negative influences from subtle rotation, translation and distortion, each feature point will be assigned with descriptors in form of position-dependent histogram (Fig. 2-13), so-called Histogram of oriented Gradients (HoG). It could not only describe the maximal gradients in one feature point, but also stays robust characteristics in case of objects' movement. In consequence, it is suitable to choose SIFT to track certain objects even though the surroundings always change, or interested objects move with speed. (Fig. 2-14)

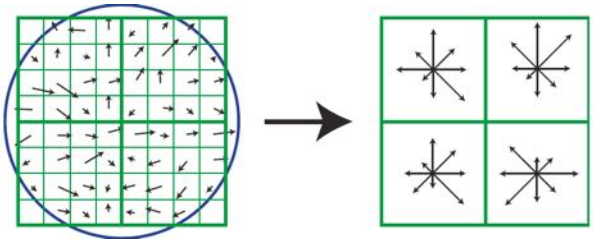


Figure 2-13: Histogram of oriented Gradients[53b]
left: Image gradients; right: HoG

Figure 2-14: Robustness of SIFT features. Feature points can still be tracked even though the camera was moved and rotated and the background became more complicated.[54]

It is worth mentioning that, SIFT get disadvantage in time-wasting or latency due to the huge computational task.

[53] Lowe, D. G.: Distinctive image features (2004), a: p. 4; b: p. 11.

[54] Kaehler, A.; Bradski, G.: Learning OpenCV 3 (2016), p. 540.

SURF

In computer vision, Speeded Up Robust Features (SURF) is a local feature detector and descriptor[55]. Actually, SURF is partly inspired by the SIFT. It is also a feature vector derived from receptive-field-like responses in a neighbourhood of an interest point. SURF descriptor does, however, differ in the following respects: (1) it is based on Haar wavelets instead of derivative approximations in an image pyramid, (2) the interest points constitute approximations of scale-space extrema of the determinant of the Hessian instead of the Laplacian operator, (3) the entries in the feature vector are computed as sums and absolute sums of first-order derivatives $\sum L_x$, $\sum |L_x|$, $\sum L_y$, $\sum |L_y|$ instead of histograms of coarsely quantized gradient directions[56].

Here, the Haar wavelet is a sequence of rescaled "square-shaped" functions which together form a wavelet family or basis. Wavelet analysis is similar to Fourier analysis in that it allows a target function over an interval to be represented in terms of an orthonormal basis. Briefly, its scaling function can be described as Eq. (2-18)[57].

$$\varphi(t) = \begin{cases} 1, & 0 \leq \mathrm{t} < 1 \\ 0, & \text{otherwise} \end{cases} \tag{2-18}$$

Experimentally, the standard version of SURF is several times faster than SIFT and claimed by its authors to be more robust against different image transformations than SIFT.

FAST

Features from Accelerated Segment Test (FAST)[58a] has similar detection method as Harris Corner Detector, but it improves the calculating speed by selecting fewer comparison pairs. First we have a pixel point P, and then compare if the "circle" around P darker or brighter. If more than half (or 3/4) of points in the circle segments are darker or brighter than value of P (surpassing threshold t as a reference), P is selected as one feature point. In order to calculate faster, the points on circle can be fewer selected.

Although it has benefits for calculating efficiency, it is vulnerable to noise or wrongly focuses on unimportant objects. Besides, it does not guarantee scale- or rotation invariant characteristics as SIFT or SURF. Therefore it is useful for certain circumstance.

BRIEF

Instead of HoG, Binary Robust Independent Elementary Features (BRIEF) uses binary system (0 or 1) to describe the comparison results of value. After selecting interest points, these points are randomly compared in brightness (Fig. 2-15) and give statistics for descriptors. The binary code system has benefit of the quicker computing and storing data

[55] Bay, H., Tuytelaars, T.,; Van Gool, L.: SURF: Speeded Up Robust Features (2006).

[56] Lindeberg, T.: Scale Invariant Feature Transform (2012), p. 11.

[57] Elsevier B.V.: Haar Function (2021).

more efficiently, and comparison speed is better than HoG. Different from functions of HoG in SIFT and SURF, BRIEF is not scale-invariant .

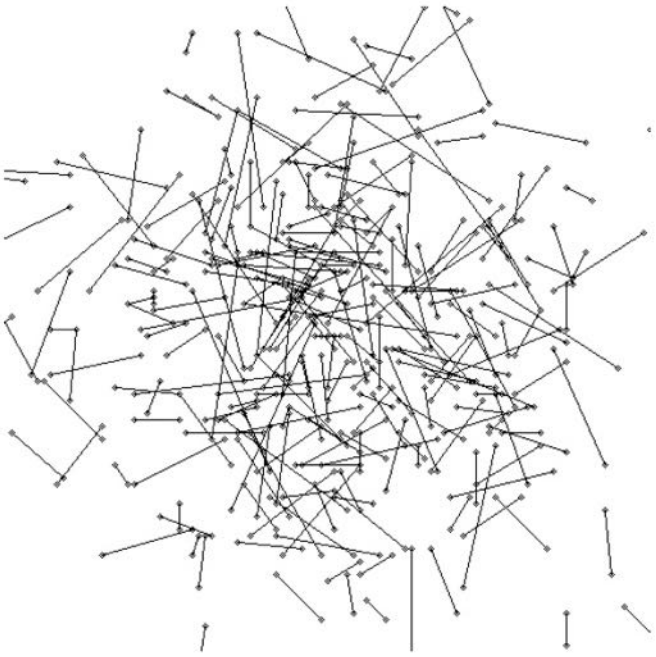


Figure 2-15: BRIEF descriptors.[58b]

ORB

Oriented FAST and Rotated BRIEF (ORB) literally combines algorithms of FAST (for detecting feature points) and BRIEF (for descriptors). We have mentioned the quickness and efficiency of FAST and BRIEF methods, so ORB can also offer a faster computing speed as SIFT or SURF. However, neither FAST nor BRIEF has scale-invariant robustness. Therefore ORB calculates orientation of points from gradients around points.

Among all these feature detection methods mentioned above, SURF, SIFT and ORB are most commonly applied for image stitching in OpenCV. Comparing these 3 methods from the calculation speed and robustness characteristics, we can obtain the conclusion shown in Table 2-3.

Table 2-3: Ranking different feature points methods with -(poor), +(normal), ++(good), +++(very good)

Methods	ORB	SURF	SIFT
Calculation speed	+++	++	+
Rotational robustness	+	++	+
Scale-invariant robustness	–	++	+

[58] Kaehler, A.; Bradski, G.: Learning OpenCV 3 (2016), a: p. 537; b: p. 555.

Homography

Hypothesizing that the original images are captured by a camera rotating about its optical center or that the scene is approximately planar, conventional image stitching algorithms focused on obtaining global 2D transformations to align one image with the other[59,60]. Briefly, this single global transformation is named as homography. A homography transforms vectors from one plane to another one[61]. In the field of computer vision, any two images of the same planar surface in space are related by a homography[62]. The homography matrix is a 3×3 matrix but with 8 degrees of freedom as it is estimated up to a scale. It relates the pixel coordinates in the two images as is shown in Fig. 2-16. And the matrix form of homography is written as Eq. (2-19)[61].

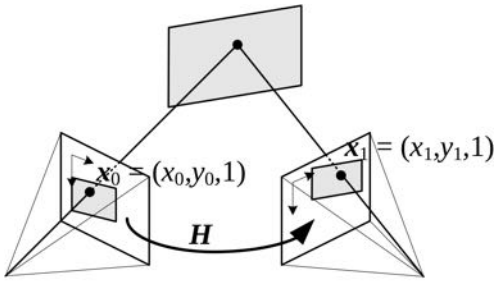


Figure 2-16: A planar surface viewed by two camera positions.(Adapted from [63])

$$s \begin{pmatrix} x_1 \\ y_1 \\ 1 \end{pmatrix} = \mathbf{H} \begin{pmatrix} x_0 \\ y_0 \\ 1 \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{pmatrix} x_0 \\ y_0 \\ 1 \end{pmatrix} \tag{2-19}$$

Here, s is a scale factor, and homography matrix $\mathbf{H}$ is generally normalized with:

$$h_{33} = 1 \tag{2-20}$$

or:

$$h_{11}^2 + h_{12}^2 + h_{13}^2 + h_{21}^2 + h_{22}^2 + h_{23}^2 + h_{31}^2 + h_{32}^2 + h_{33}^2 = 1 \tag{2-21}$$

A common application of global transformation is Bird-View visualization, which is used in automotive park system. The generation of a Bird-View visualization is based on the assumption of a “flat world”. Under this assumption, the world is supposed to be flat, thus providing that all objects which are imaged by a camera lie on the same plane. Assuming calibrated cameras, the original images can be projected onto this plane, allowing for

[59] Szeliski, R.: Image alignment and stitching (2007).

[60] Brown, M.; Lowe, D.: Automatic panoramic image stitching (2007).

[61] OpenCV: Basic concepts of the homography (2021).

[62] Wikipedia contributors: Homography (computer vision) (2020).

[63] Szeliski, R.: Computer vision: algorithms and applications (2010).

a change of perspective by means of a virtual camera. The virtual camera is usually positioned fixed on top of the vehicle, although there is no real restriction in this sense[64].

Parallax

In practice, many images to be stitched are very different from the standard data sets (i.e. the scene is a roughly planar, or images are captured by purely rotating the camera about its optical center). This will probably lead to misalignments and ghosting effects. In fact, these challenges are caused by different camera spacings (i.e., baselines) of the images. Baseline, which means as the distance between two viewpoints, is generally divided into 3 degrees. A median distance between adjacent images of 0.8 m is regarded as a natural baseline, 1.6 m is regarded as a wide baseline, and 2.4 m is regarded as a very wide baseline[65]. Wide baseline can absolutely lead to large parallax shown in Fig. 2-17, which is a displacement or difference in the apparent position of an object viewed along two different lines of sight[66].

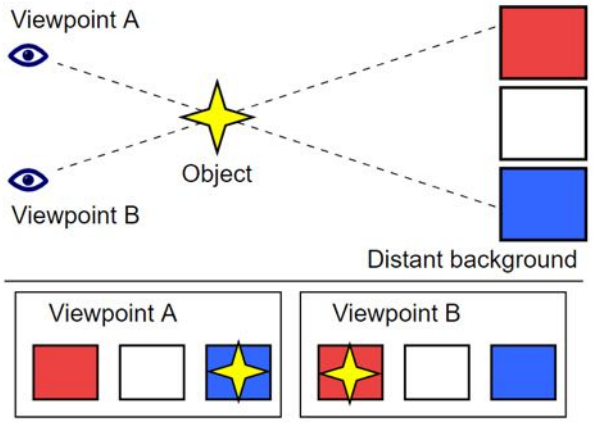


Figure 2-17: A simplified illustration of the parallax.[66]

Since parallax fades away as the distance to the object increases, and it only disappears when the distance is infinite or the baseline is zero, large parallax is likely to cause mismatching and misalignment when the objects in the images are located at different distances from cameras. Arguably, most of the problems in 2D image stitching happen because it is impossible to estimate the stitching field accurately due to the complex interaction between the 3D scene and the camera parameters. Consequently, image stitching can be much more complicated by the introduction of parallax, which is still a tough challenge in many practical applications.

Since the final goal of image stitching is to seamlessly blend overlapping images, even in the presence of parallax or lens distortion, to provide a mosaic without any artifacts

[64] García, J.: 3D Reconstruction for Optimal Representation (2015).

[65] Flynn, J. et al.: DeepStereo: learning to predict new views (2016).

[66] Wikipedia contributors: Parallax (2021).

that looks as natural as possible. In recent years, several assumptions can be posed on the stitching field during image alignment[67] and tolerance to parallax can also be imposed.

To satisfy more complicated applications and solve the issue of aligning the images containing multiple planes in the overlap regions, these optimised methods calculate local hybrid transformation rather than a single global homography matrix. An image is first divided into uniform grids, and each grid is warped and aligned with a homography estimated by introducing a Moving Direct Linear Transformation (Moving DLT) method in As-Projective-As-Possible (APAP)[68]. Motivated by APAP, several researchers have tried to stitch challenging data sets with parallax and baseline by introducing mesh-based alignment optimization algorithms to improve the initial alignment. Zhang and Liu[69] first roughly align the input images by performing SIFT, and then construct an objective function consisting of alignment and some prior constraint terms to further optimize the alignment. As an alternative solution for our project, these advanced methods will be introduced in detail later in Chap. 4.

[67] Lin, w.-y. et al.: Smoothly varying affine stitching (2011).

[68] Zaragoza, J. et al.: As-projective-as-possible image stitching with Moving DLT (2013).

[69] Zhang, F.; Liu, F.: Parallax-Tolerant Image Stitching (2014).

3 Requirements Analysis

In this chapter we will discuss the requirements of this ADP. We are considering three types of requirements. There are the **tool** requirements, which in the end determine which software and hardware tools are eligible. Further, we have functional requirements that make demands which must be fulfilled by the **result** of the program. Last but not least, we also distinguish into environmental restrictions which are directly coupled to the task or the **environment** respectively.
Table 3-1 displays the requirements for this work. Since the MAAS project software is based on ROS Noetic Ninjemys[70], it is also mandatory to implement the necessary function into a ROS node. Additionally, ROS supports the programming languages C++ and Python 3, which means either of those has to be used for the implementation.
Like we have already stated in Sec. 1.1, MAAS stands for and deals with the automation of trams in a traffic environment. This leads, especially in the teleoperation case, to tight requirements on the computation time which contributes to the total latency. In detail, this has been discussed in Subsec. 2.2.1.
Besides the prerequisites framing the software module, Table 3-1 also shows the functional requirements describing the demanded result. Firstly, the image size must be consistent enable further processing and guarantee an appropriate experience for the operator. Second, the frame rate of the output video shall be at least 22 Frames per Second (fps) such that the video stream makes a fluent impression to a human operator[71]. The operator work space features a Samsung LC49RG94SSUXZG flat screen monitor with a resolution of 5120×1440 pixel[72]. Therefore, it is a waste of computation time to produce a result that exceeds this resolution and thus, has to be avoided. Although the resolution has an upper bound, the image quality shall very as possible to preserve as much detail as possible to give the operator all the information they need to operate the tram from afar. In addition, the resulting images in the video stream should lack visible image boundaries and have a smooth transition between the original images and have a consistent brightness. The stitched images may also be free of ghost and the objects therefore are supposed to have hard contours. Lastly, in the output video frames straight lines should be displayed as straight lines[73] to give the operator a natural feeling and improve the immersion over distorted images.

[70] https://wiki.ros.org/noetic.

[71] Luckas, V.: Mensch-Maschine-Kommunikation – Wahrnehmung (2015), p. 22.

[72] Samsung Electronics GmbH: datenblatt LC49RG94SSUXZG (2019), p. 1.

[73] Only if that is applicable.

Table 3-1: Project requirements

F/W	Name	Value	Description	Resource
Tools				
F	System framework	ROS	MAAS System is based on ROS	System
F	Programming language	C++ or Python 3	ROS supports C++ and Python	ROS
Result				
F	Image size	smaller than 5120×1440 Pixel	Resolution higher than screen resolution is waste of resources	Hardware
F	consistent size		Image size does not vary	System
W	Image	Quality as high as possible	Details are preserved	
		Straight lines are kept as straight lines	Operator needs a natural feeling	
		No visual image boundaries	Smooth image without brightness jumps	
		No Ghosts	Clear image	
F	Frame rate	min. 22 fps	fluent video	Human body
Restrictions				
W	Total run time	smaller than 100 ms	Latency must be as small as possible	Environment System

4 Solution Concepts

In this section we will present and discuss some alternative approaches to solve the task of our ADP, i.e. a method to stitch video streams of various cameras with rectilinear and fisheye lenses. Since we have already explained some important theories in Chap. 2, which are related to our project, next we will focus on specific operational methods used for distortion correction and image stitching process.

4.1 Solution Concepts for Fisheye Rectification

As is clarified in Sec. 2.4, the main approach to solve the fisheye lens correction is to use the camera matrix and the distortion coefficients, which are obtained by the calibration process. In detail, there are several functions used for elimination of image distortion in OpenCV. For instance, we can simply choose function `cv::undistort()`, or we can first use `cv::getOptimalNewCameraMatrix()` to obtain new camera matrix. Then we call `cv::initUndistortRectifyMap()` get the mapping relationship of the coordinate axis. Finally we apply `cv::remap()` to map the original image to the target image. Actually, both of these two methods are essentially based on the same principle. It finds the corresponding original image coordinates, and then copies its value to the target image. Due to the image size and transformation, interpolation or approximation methods are required, such as nearest neighbor method, linear interpolation, etc. While the function `cv::undistort()` also calls `cv::initUndistortRectifyMap()` and `cv::remap()` during the running process, for every new image we use this function to eliminate distortion, it needs to recalculate the mapping relationship. This wastes so much time that probably causes delay for later process. However, if we put the function `cv::initUndistortRectifyMap()` outside the loop, it will only calculate mapping for the first image, and following images can use the same mapping.

From the current engineering reality, although the existing fisheye image distortion correction algorithms can achieve distortion correction to varying degrees, there is loss of image edge information and partial image shortcomings such as enlargement. The image in the center of the traditional spherical projection correction image is very clear, and the surrounding scenes are blurred and incomplete. The reason is that the image is stretched toward the surroundings, similar to being enlarged, forming a blurred surrounding image. Although many researchers have developed calibration algorithms of radial distortion of fisheye lens, quantitative evaluation of the correction performance has remained a challenge[74].

Considering that the output of this step has to be processed further, i.e. being stitched together to a single frame containing all the information from all cameras , what we value most is the overall effect after fusion. In addition, we also need to pay attention to the time-consuming problem of the entire algorithm. Since iterative method has a relatively large amount of calculation, it takes a long time and needs to be optimized in the follow-up.

[74] Liu, Y., Tian, C.,; Huang, Y.: Critical Assessment of Correction Methods (2016).

4.2 Image Stitching Using Stitching Pipeline

After all target requirements were achieved and the distortions of each images rectified, they can be stitched into a panorama picture to attain our final result in this project. We will first introduce the stitching pipeline method provided by OpenCV. This stitching algorithm consists of steps: image registration(feature detection, feature matching), image matching, bundle adjustment, wave correction, surface warping, seam finding, exposure compensation and image blending. As an idiomatic method, his entire process is shown in the figure below.

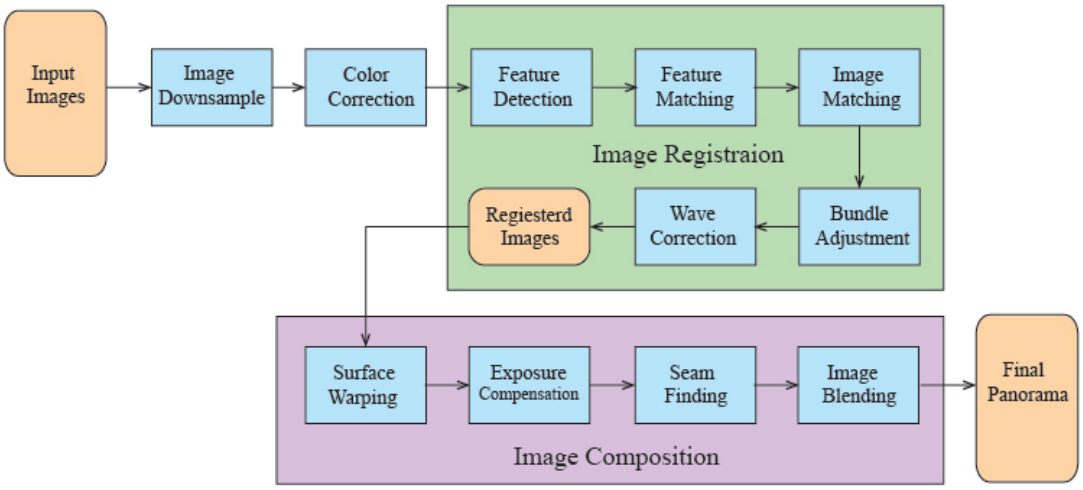


Figure 4-1: Stitching pipeline.[75]

Feature Detection

Literally, feature points are some meaningful parts in one photo which provide symbolic information in the photo so that the objects in it could be detected. In Subsec. 2.5.2, we have introduced different feature extractors and descriptors. When the feature points are detected, the amount of data can be reduced.

Feature Matching

After extracting features from all images, we should match them. First it is intuitive to use brute force matching. This method matches every feature point of the first image with every feature point in the other image. However, it causes waste of time. Wrong matches are also problem if there are duplicate objects appearing in pictures. In other words, two different objects with same characteristics would be seen as one thing. It is useful to adjust parameters called cross-checking in order to restrict compared regions in pictures. However it costs extra computing time.

[75] Kim, J.: Panoramic Image Communication for Mobile Application (2017), p. 340.

Another method is Fast Library for Approximate Nearest Neighbor (FLANN). Since multiple images may overlap a single ray, each feature is matched to its k nearest neighbours in feature space. This can be done by using a k-d tree or k-means to find approximate nearest neighbours. Here, OpenCV gives a library FLANN, which is a fast implementation of k-d tree. Parameters like numbers of k-d trees can be chosen. Generally, FLANN provides more efficient and correct way to match feature points than Brute Force, so it is usually applied.

Regardless of which of the two matchers is used, to further improve the matches one filters the matches with a method known as Lowe's ratio test which has been introduced with SIFT[76].

The idea is to eliminate all feature points where the best and second best match are approximately equal in terms of quality. To do so, one uses a k-Nearest Neighbors (k-NN) matching with $k = 2$ to yield the first and second best match (m_1 and m_2). [76] Based on the feature descriptor used, one will apply a different distance measure between the matched feature of the first and second image. We will denote this distance operator as $||\cdot||$. We only keep the features for which

$$||m_1|| < r_l \cdot ||m_2|| \tag{4-1}$$

holds true. The others are filtered out.[76] We will call r_l Lowe's feature ratio.

Figure 4-2 shows the result of a feature matching. In OpenCV this function is called `drawMatches`, and colors can also be regulated by ourselves.

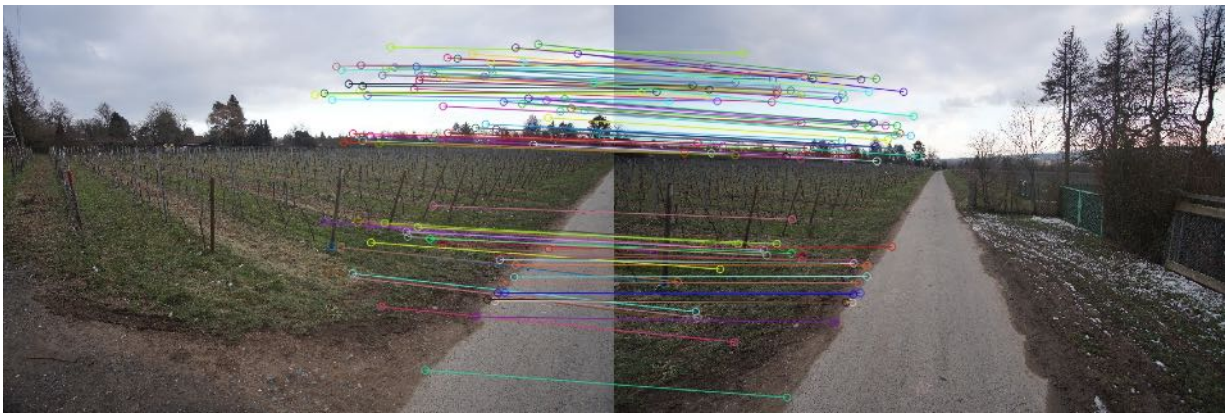


Figure 4-2: Feature matching.

Image Matching

At this stage, the goal is to find all matching (i.e. overlapping) images. The connected image matching set will later become a panorama. Because each image may match each other, the problem at first seems to be quadratic in the number of images. But obviously it is only necessary to match each image with a small number of overlapping images to obtain a good solution to the image geometry. In order to get pairwise matches between

[76] Mordvintsev, A.; Revision, A. K.: Feature Matching – OpenCV-Python Tutorials 1 documentation (2013).

images, first we use Random Sample Consensus (RANSAC) to select a set of inliers that are compatible with a homography between the images, then we apply a probabilistic model to verify the match.[77a]

Robust Homography Estimation Using RANSAC Random Sample Consensus (RANSAC) is a robust estimation program that uses a minimum set of correspondences of random samples to estimate image transformation parameters and finds the solution with the best consensus with the data. RANSAC is essentially a sampling method for estimating homography.[77b] Figure 4-3 shows the difference without (a) and with (b) the use of RANSAC. Incorrect feature pairs are removed when using RANSAC.

(a) (b)

Figure 4-3: Comparison with and without RANSAC.

Probabilistic Model for Image Match Verification For each pair of potentially matched images, we have a set of geometrically consistent feature matches (RANSAC inliers) and a set of features that are in the overlap area but not consistent (RANSAC outliers). The idea of our verification model is to compare the probability that this set of internal inliers/outliers are generated by correct image matching or incorrect image matching.Using this method, we can find the panoramic sequence as a connected set of matching images. This allows us to identify multiple panoramas in a set of images and reject noisy images that do not match other images[77c].

Bundle Adjustment

Given a set of geometrically consistent matches between images, bundle adjustment is an indispensable step, because the cascade of pairwise homographies would cause accumulated errors and ignore multiple constraints between images. Through bundle adjustment, all camera parameters can be solved together.[77d]

Wave Correction

The image registration using the previous steps only gives the relative rotation between each best matching pair, assuming that the 3D rotation to the selected world coordinate is the identity matrix. This assumption will result in a wavy effect in the final panorama, especially when we have large amount of input images. The idea of wave correction is based on the fact that although people may tilt and rotate the camera when taking images, they rarely twist the camera relative to the horizontal plane. Therefore, by finding the nill

space of the camera horizontal plane covariance, the global rotation matrix can be found to eliminate the wave effect.[77e]

Surface Warping

After the images are aligned correctly, they will be mapped to the stitched surface. With OpenCV the flat, cylindrical and spherical surface warping method can be easily realised. Once the surface warping is completed, the overlapping pixels between each pair can be calculated.[77f]

Seam Finding

If the images are stitched sequentially, object motion and spatial alignment errors will cause ghosting artifacts. Therefore, an optimal seam finding method is needed to find seams in the overlapping area of the source image, create labels for all pixels in the composite image, and stitch the source image along the optimal seam.[77g]

Exposure Compensation

Beside the geometric parameters of each camera, in order to achieve visual appealing panorama result, we also need to find the photometric parameter of each camera, which is the overall gain of each pairs. The goal is to minimize the sum of gain normalized intensity errors for all overlapping pixels.[77h]

Image Blending

We want every pixel along the ray would have the same intensity in every image it intersects, however, even after exposure compensation, some image edges are still visible due to many unmodeled effects, including vignetting (decreases of the intensity toward the edge of the image), parallax effects due to undesired movement of the optical centre, misregistration errors due to mismodelling of the camera, radial distortion and so on[77i]. OpenCV's Multibind blending is a very effective method.

4.3 Image Stitching Using AANAP

Many methods only focus on global 2D photo stitching, which could lead to misalignments and ghosting effects if we consider different perspectives between moving cameras (3D photo stitching). Therefore, it is necessary to apply more complicated methods to achieve the goal, such as Adaptive As-Natural-As-Possible (AANAP)[78] image stitching method.

[77] Brown, M.; Lowe, D.: Automatic panoramic image stitching (2007), a-b: p. 2; c-d: p. 3; e-i: p. 7.

[78] Lin, C. et al.: Adaptive as-natural-as-possible image stitching (2015).

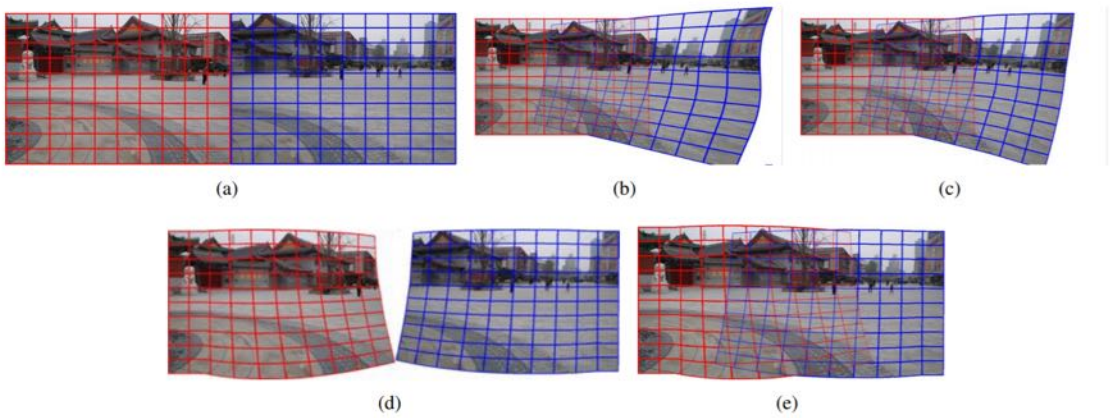


Figure 4-4: AANAP Process: (a) Original images (b) Warp after applying moving DLT with Gaussian weighting (c) Extrapolation of non-overlapping areas using homography linearization (d)Final warp after global similarity transformation (e) Final stitched image.[78]

AANAP actually consists of processes and we introduce it as below. First we prepare two original images (Fig. 4-4(a)), then the overlapping part need to be stitched using a method called As-Projective-As-Possible (APAP)[79], which calculates local homography in the overlapping area to avoid perspective distortion. In addition, a simple moving Direct Linear Transformation (DLT), as known as Gaussian weighting, is used to obtain seamless results in images, by providing higher weights to closer feature points and lower weights to the farther ones. Because of homography partitions, the problem of small parallax is well solved, however APAP sacrifices distortion outside the overlapping part, as seen in Fig. 4-4(b). By linearizing homography in non-overlapping part, the phenomenon could be avoided. (Fig. 4-4(c))

Although the result is well improved, the right outside scene of panorama could not well be aligned with left scene. The solution in AANAP finds similarity transformation in the overlapping area, so that two previously warped images have similar scales and perspective angles (Fig. 4-4(d)). Finally the stitched photo has more natural presentation in every pixels (Fig. 4-4(e)). It should be noted that the calculating of similarity transformations bases on local partition of areas because of different perspectives between two images.

AANAP presents a more natural panorama without visible parallax in the overlapping area and avoids perspective distortion in the non-overlapping area. However it could not solve this problem well, when the baseline of cameras is too wide. More future researches should focus on this challenge.

4.4 Image Stitching Using NISwGSP

Although AANAP supply satisfying results for image stitching, the drawback could be magnified when more images get stitched. The reason is that AANAP focuses only on local warp model when calculating similarity transformation. In this case scale and rotation

[79] Zaragoza, J. et al.: As-projective-as-possible image stitching with Moving DLT (2013).

parameters (especially rotation) would be easily ignored and cause the "curve" in panorama stitched image (Fig. 4-5 (top)).

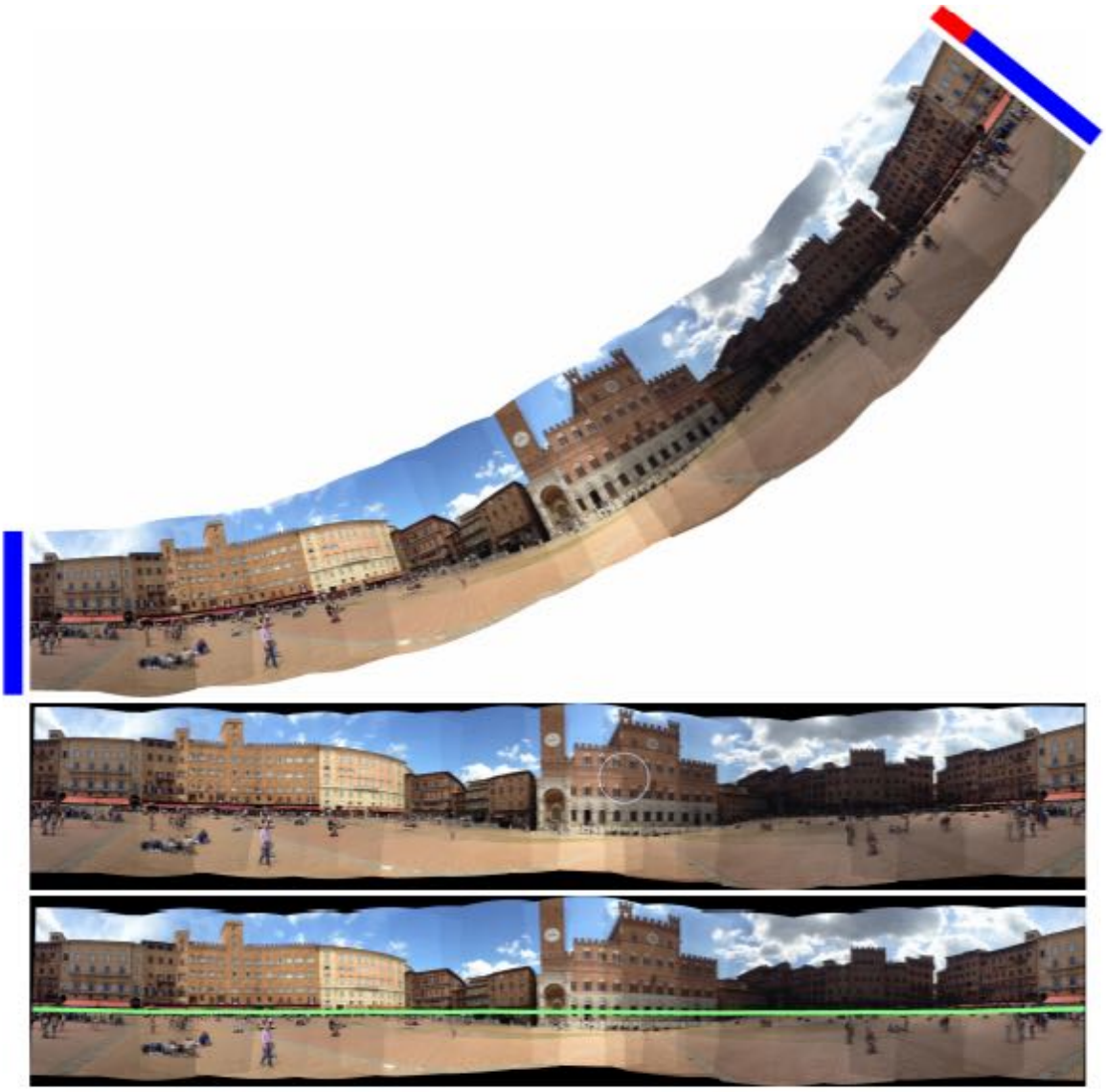


Figure 4-5: Multi-Image-Stitching: AANAP (top), NISwGSP (middle), NISwGSP with a specified horizon line (bottom).[80]

To solve the problem we need to select suitable scale and rotation parameters, so that the similarity transformation could be globally calculated. That is exactly thought of Natural Image Stitching with the Global Similarity Prior (NISwGSP). The former procedure in NISwGSP resembles AANAP as well, but adds Global Similarity Prior (GSP) in last part. GSP is a process with scale and rotation selection. First, the rotation angle can be determined from homography by APAP (former procedure in AANAP). If only two images await stitching, then similarity transformation takes next step. In case with

[80] Chen, Y.-S.; Chuang, Y.-Y.: Natural Image Stitching with the Global Similarity Prior (2016), p. 3.

multi-image, the rotation angles from different pairs of images should be gathered, and Minimum Line Distortion Rotation (MLDR) decides which rotation parameter is chosen. This procedure guarantees multi-image-stitching has an average, global rotation angle so that the panorama image straightens properly (Fig. 4-5 (middle)). Also the result can be optimized by concerning about one specified horizon line (Fig. 4-5 (bottom)).

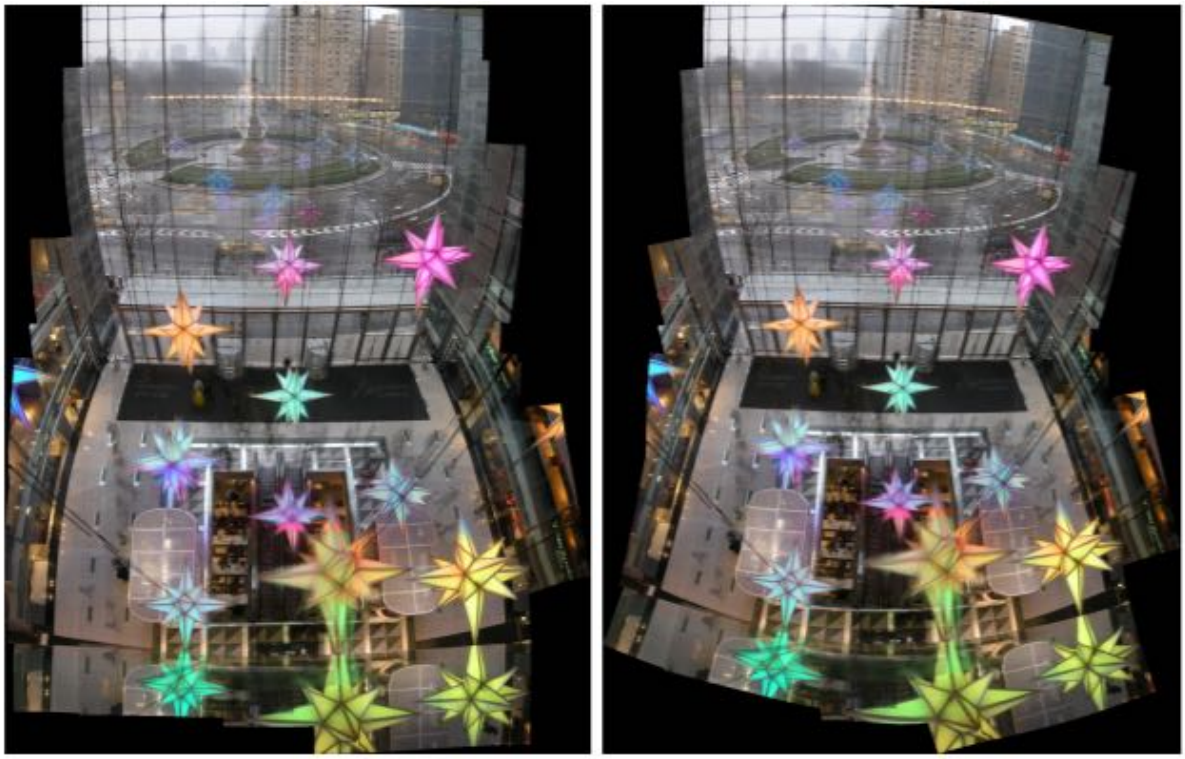


Figure 4-6: NISwGSP. 2D-Method (left), 3D-Method (right).[81]

There are also two different method to calculate MLDR. One is 2D-method, which is used as the camera angle does not twist too far away from horizontal line. The other is 3D-method, used as stitched images are in a wide range (x and y axles) instead of along horizontal line (x axle). As seen in Fig. 4-6, 2D-method performs well in aspect of two neighbor photos but has obvious twisted line in global view, while 3D-method performs better in global view.

4.5 Image Stitching Using PtIS

The Parallax-tolerant Image Stitching (PtIS)[82] method presents a local stitching method to handle parallax based on the observation that input images do not need to be perfectly aligned over the whole overlapping region for stitching. Instead, they only need to be aligned in a way that there exists a local region where they can be seamlessly blended together. It adopts a hybrid alignment model that combines homography and content-preserving warping to provide flexibility for handling parallax and avoiding objectionable

[81] Chen, Y.-S.; Chuang, Y.-Y.: Natural Image Stitching with the Global Similarity Prior (2016), p. 14.

[82] Zhang, F.; Liu, F.: Parallax-Tolerant Image Stitching (2014).

local distortion. We predict how well a homography enables plausible stitching by finding a plausible seam and using the seam cost as the quality metric. An example stitching result of PtIS is shown in Fig. 4-7. The two image at the top are the input image and the image (c) at the bottom is the stitched result.

(a) Input image 1

(b) Input image 2

(c) Stitched image

Figure 4-7: PtIS Stitching example.[83]

4.6 Potential Solutions

Numerous studies have shown that the challenges of wide baseline, large parallax, and low-texture overlapping regions mainly result in the inability to obtain enough matched feature pairs, further misregistration for stitching, or ghosting and blurring in the stitched views. In such cases, methods from other fields could be adopted, such as 3D reconstruction and Simultaneous Localization and Mapping (SLAM), and point cloud-based methods are introduced to register the images. In this part, we present some stitching solutions proposed by the innovation of different technologies.

4.6.1 Deep Semantic Feature Matching

In recent years, the superior performance of deep learning has been proved and many studies focus on the semantic contents of images with learned Convolutional Neural Network (CNN) features. Owing to their invariance to geometric deformations and changes in illumination, CNN features accurately locate the salient features. The Deep learning-based semantic flow algorithm[84] adopts pre-trained CNN features to build a feature pyramid of each image and selects salient features for matching on different scales

[83] Zhang, F.; Liu, F.: Parallax-Tolerant Image Stitching (2014), p. 4.

[84] Ufer, N.; Ommer, B.: Deep Semantic Feature Matching (2017).

using informatics criterion on the cell activations of the pyramid. For each selected feature a set of matching candidates is extracted and the final assignment is obtained by solving an energy minimization problem with a unary appearance and a binary geometric term. Long-range contextual relationships are preserved by a fully connected graph. To improve results in real-world images without bounding box annotations, additional unary and binary objectness potentials are introduced. Finally, a dense flow field from the sparse correspondences using Thin Plate Splines (TPS) was estimated. The overview of approach for dense semantic matching is shown in Fig. 4-8.

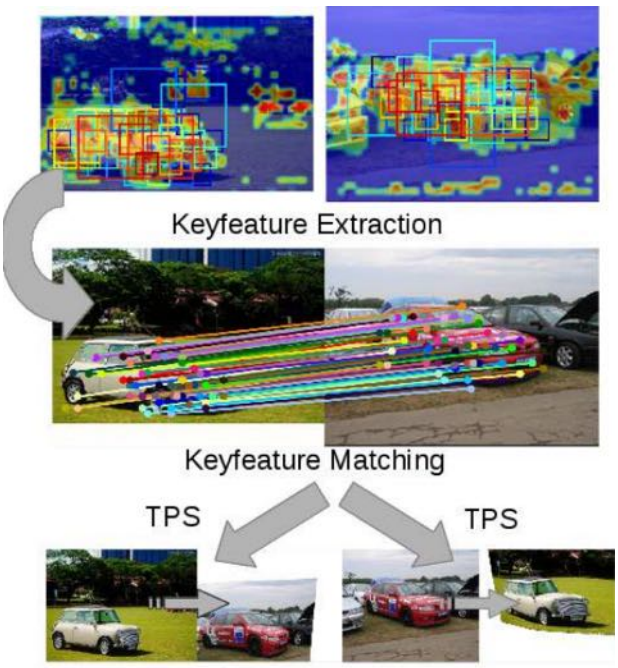


Figure 4-8: Deep semantic feature matching.[84]

According to this deep semantic feature matching, it is possible to stitch all the camera views into one panoramic video stream based on deep learning. However, these networks are trained with what computer scientists would call "natural images" with very limited barrel distortion (shown in Fig. 2-6(a)), e.g. taken by a standard zoom lens. The rectified images differ from those training data and it is well known that Neural Networks cannot extrapolate very well. Another idea, is to train a Neural Network (NN) with a sophisticated CNN architecture end to end which directly learns to stitch the video stream from the MAAS tram. Unfortunately, this approach is not only difficult to design and train but also needs a lot of training examples which do not exist. Although these examples can be created from the sensor output with an image processing software like GIMP[85] or Adobe Photoshop[86], the amount of work would be unbearable and therefore vastly exceed the frame of this ADP or even MAAS itself.

[85] https://gimp.org.

[86] https://adobe.com/photoshop.

In order to deal with different perspectives from many images, local warp models are chosen as a solution. However it fails with big parallax . As above explained, CNN could be a suitable training model and solves deficiency of depth data for 3D-scenario. The method Image Stitching Based on Planar Region Consensus[87] offers concrete models for CNN training: Because the same objects have similar RGB value even in two different photos, we could cluster these pixels as a consistent planar region based on RGB value and send as a training sets for CNN. Thus, It is easier to find correct feature points according to the planar region. More outliers could be automatically eliminated if they do not belong to same group.

4.6.2 3D Modelling and Stitching

To effectively stitch the images with very wide baseline and very large parallax, a novel method of multiple video fusion in 3D environment was proposed by Zhou[88a], which produces a highly comprehensive imagery and yields a spatio-temporal consistent scene. This 3D stitching method is based on 3D reconstruction from a single image through estimating reference camera poses and 2D image matching information to align their adjacent 3D models. Matched point pairs are added into the intersecting plane space of 3D models, and dense mesh vertices are constructed with the added keypoint pairs, to further deform the original images. Next, the aligned images are blended from a virtual viewpoint. In this matching process, not only the pose parameters of cameras and reconstructed models are required, but also the 3D models do play an important role. As is shown in Fig. 4-9, the entire process can be divided into offline and online, the former includes automatic pre-processing, interactive modeling, and 3D stitching. The latter combines real-time rendering technology to achieve multi-video fusion.

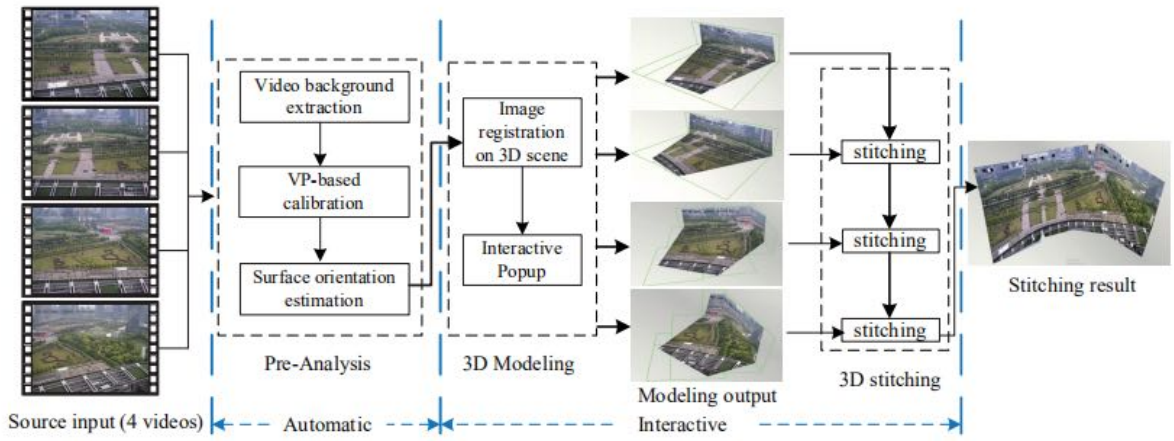


Figure 4-9: Process of interactive 3D modeling and stitching.[88b]

This method performs theoretically on a suitable depth plane, and each image should be segmented into multiple sub-regions according to different depths, which are caused

[87] Li, A., Guo, J.,; Guo, Y.: Image Stitching Based on Planar Region Consensus (2020).

[88] Zhou, Y. et al.: MR Video Fusion: Interactive 3D Modeling and Stitching (2018), a: -; b: p. 3.

by large parallax or wide baseline. Finally, the stitched result can be observed at any viewpoint without a large distortion as Fig. 4-10.

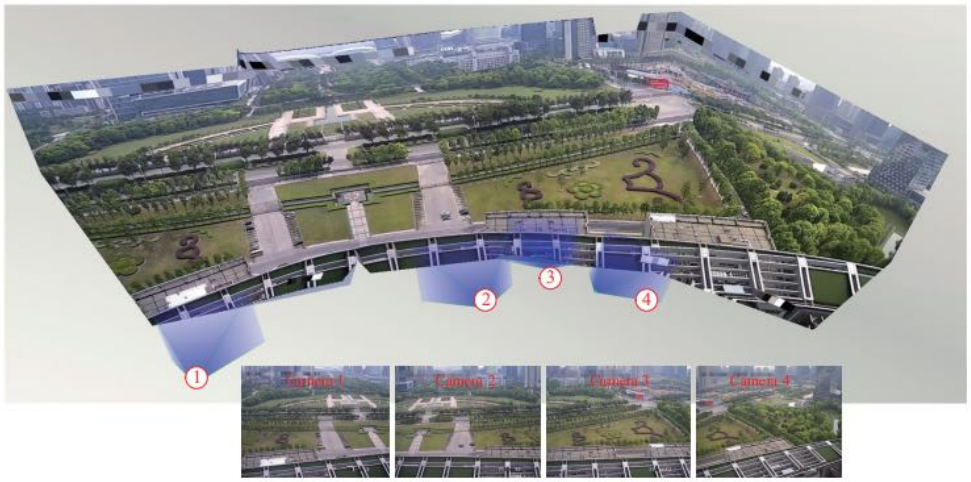


Figure 4-10: Mixed Reality Video Fusion. Bottom: Four input videos with wide baselines, while the blue frustum indicates the cameras' locations and orientations; Top: 3D stitching result based on modeling method.[89]

Meanwhile, some disadvantages cannot be ignored. Firstly, the 3D reconstruction is only conducted in the intersecting plane space and a suitable automatic depth estimation method is not available yet. Furthermore, the final 3D scene is really flexible, so that distortion appears while observer is rotating the angle of view. Last but not least, a large latency can be caused during the whole stitching process. As is mentioned above, this method is divided into offline and online parts. Hence, the ideal target of our ADP is to stitch all the camera views into one real-time panoramic video stream. Since the environment situation changes with the tram's movement, it is significant to keep the network stable all the time and reduce the latency as much as possible during the teleoperated driving process.

4.6.3 Vanishing Point Guided Natural Image Stitching

This method considers the guidance of vanishing points to solve the severe projective distortion and unnatural rotation. Because the mutually orthogonal vanishing points in the Manhattan world can provide really useful directional clues, this method designed a scheme to effectively estimate image similarity. Impressive natural stitching performance can be achieved by putting a prior information such as global similarity constraints into the popular mesh deformation framework.[90]

[89] Zhou, Y. et al.: MR Video Fusion: Interactive 3D Modeling and Stitching (2018), p. 1.

[90] Chen, K. et al.: Vanishing Point Guided Natural Image Stitching (2020).

5 Solution Selection and Implementation

In this part, we will discuss the advantages and disadvantages of the aforementioned approaches. Based on the importance of the criteria, discussed first in this part, we will decide on a solution concept. Further, this section also includes our findings on the application and implementation of these approaches.

5.1 Selection Criteria

In the following, we will present the most important selection criteria and their importance and weight. They are displayed in Table 5-1. The computation time or speed is basically the most important criteria when it comes to choosing a solution method. The requirement is to develop an almost real time application with minimal delay to ensure the safety the passengers and other traffic participants close to the vehicle. Besides the significance of the run time, memory complexity plays a minor role. The memory of the two computers involved should be sufficient to cover the needs and can be traded off for shorter execution time. Memory these days is cheap compared to computing power and can be easily extended if necessary. Further, the image quality is also of very high importance. Since the visual input is the main source of information the operator gets, it must satisfy his needs to immerse into the scene. Our team consists mainly of mechanical engineers with very little experience in programming and no experts in code optimization. Thus, the simplicity of the implementation cannot be left out of sight and must be considered when choosing a concept. Last but not least, there the software could be modular to be easily expandable and reusable. Since this project aims to solving a particular problem with very high constraints on the computation time, this criteria is negligible exceeding the modularity of a ROS node.

Table 5-1: Selection criteria with their importance ranging from –– (unimportant) over ○ (neutral) to ++ (extremely important).

Computation time	++
Image quality	++
Complexity of the implementation	+
Extensibility and Modularity	-
Memory usage: RAM and Hard drive	--

5.2 General Design

As mentioned in Chap. 3, the programming can either be implemented in Python or C++. Whereas Python is easier to learn and use, C++ runs faster. The programming language chosen is heavily based on the code provided in the papers. Mostly, for simplicity, we used the dominant programming language to develop a ROS node.

It is also possible to divide the functionality into two or more ROS nodes. Here, the trade-off is modularity versus speed. Merging of video streams is a real-time application,

because rectified and stitched images must be displayed to the operator with the least possible delay (see Subsec. 2.2.1) to allow teleoperation. Our application is intended to create a panoramic image from camera images, which is a modular function and for this reason we do not consider it necessary to split this functionality into several ROS nodes.

In addition, there are two possible entry points for the software developed in this ADP, i.e. on the tram and on the operator work station. The benefit of the operator work station, on the one hand, is has a by far superior graphic card compared to the tram. On the other hand, the former mentioned approach can deal with images of better quality which may result in a better output and possibly in a reduction of the data size that has to be sent. In order to come to a decision, it must be possible to time the entire application. However, we were not able to make a decision. This is on the one hand because no suitable algorithm for stitching could be found and on the other hand because we did not have access to a suitable GPU.

5.3 Image Synchronization

Unlike the stitching of still images, video stitching needs to ensure that the cameras capture a frame at the same time. In other words, the frames that need to be stitched from different cameras should have the same timestamp. This is the basic premise to ensure the smoothness of the seam. For this problem, ROS provides a feasible solution. The process is realized by `message_filters` package, a set of message filters which take in messages and may output those messages at a later time, based on the conditions that filter needs met. It collects commonly used message "filtering" algorithms into a common space. A message filter is defined as something which a message arrives into and may or may not be spit back out of at a later point in time. The Synchronizer filter is templated on a policy that determines how to synchronize the channels. There are currently two policies: ExactTime and ApproximateTime[91]. ExactTime policy requires messages to have exactly the same timestamp in order to match. The callback is only called if a message has been received on all specified channels with the same exact timestamp. ApproximateTime policy uses an adaptive algorithm to match messages based on their timestamp, it will automatically choose to synchronize adjacent frames without requiring that the timestamp is exactly the same. After investigating the ROS bag data, we found that there are almost always subtle differences between the timestamps of different camera frames, so we choose to use ApproximateTime policy.

[91] Open Source Robotics Foundation, Inc.: message_filters/ApproximateTime (2010).

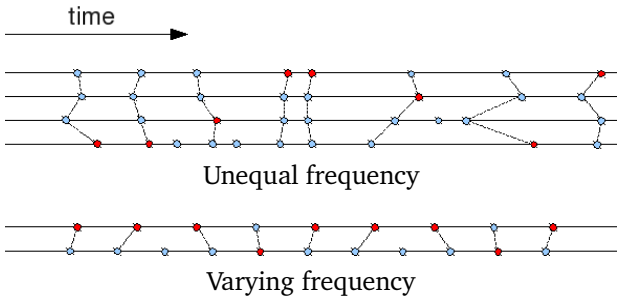


Figure 5-1: ApproximateTime policy.[91]

5.4 Time Synchronization of MAAS tram and Operator Workstation

The MAAS tram and operator station each have a computer, and the ROS node running on it ensures the progress of video transmission. It is a fact that computers do not keep time very well. The time of computers is generally based on the oscillation of crystals which are based on the frequency in the line voltage. The line voltage frequency varies from the normal $\pm$ 60 cycles per second, so clocks do not match the accuracy of an atomic clock[92]. In order to calculate the delay caused by video processing and transmission between ROS nodes, we need to refer to the current time for subtraction calculations. There are two supported protocols for synchronization of computer clocks over a network. The older and more well-known protocol is the Network Time Protocol. In its fourth version, NTP is defined by IETF in RFC 5905. The newer protocol is the Precision Time Protocol (PTP), it is defined in the IEEE 1588-2008 standard. PTP was designed for local networks with broadcast/multicast transmission and, in ideal conditions, the system clock can be synchronized with sub-microsecond accuracy to the reference time. NTP was primarily designed for synchronization over the Internet using unicast, as it is currently implemented in chrony and ntp, in local networks the accuracy can get within tens of microseconds[93]. Due to experimental conditions, our ROS nodes are deployed on the same computer, so this step is omitted.

5.5 Fisheye Lens Correction

In order to create the best user experience, i.e. allow the operator on the operator workstation to immerse into the actual scene of the tram on the street, we have tested different kinds of distortion correction. We started with the standard lens correction algorithm described in Sec. 2.4. This has been done using the corresponding OpenCV functions[94]. Therefore, our supervisor has provided us the intrinsic camera parameters, namely the camera matrix and the distortion coefficients, of the three fisheye cameras mounted to the tram. These have been produced through the camera calibration process of ROS. All the parameters are listen in the Annex in Table A-3. From our subjective view, it was not evident whether the resulting undistorted output images were useful for

[92] Arts Management Systems: Time on Computer Differs From Server (2021).

[93] Lichvar, M.: Combining PTP with NTP to Get the Best of Both Worlds (2016).

[94] Jiang, K.: Calibrate fisheye lens using OpenCV – part 1 (2017).

a human operator. An example undistorted image using these parameters in shown in Fig. 5-2. Since the those images were barely useful for the stitching process, we have also done the camera calibration process again ourselves using the corresponding OpenCV functionality[94]. Since for both pairs, the basic approach[94] cuts off too much important information, we used the one that offers more freedom[95].

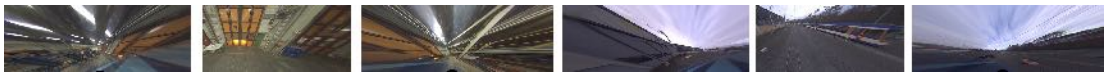
Figure 5-2: Example undistorted image of the left fisheye camera using the intrinsic camera parameters provided by our supervisor.

Besides these two parameter sets, we have also created a third set of the camera matrix based on the documentation about the camera and lens (see Subsec. 2.1.2) and detailed inspection of the fisheye images. These findings have been fine tuned to yield an output that should combine the best of both worlds, the fisheye image on the one hand, and the undistorted image with almost perfectly straight lines.

To get get an objective opinion about the visual quality for humans, we have done a survey with 55 people. For completion, we not only used these three parameter sets but also the fisheye images, as well as further processed images with close to parallel lines. All these images are shown in Fig. 5-3.

[95] Jiang, K.: Calibrate fisheye lens using OpenCV – part 2 (2017).

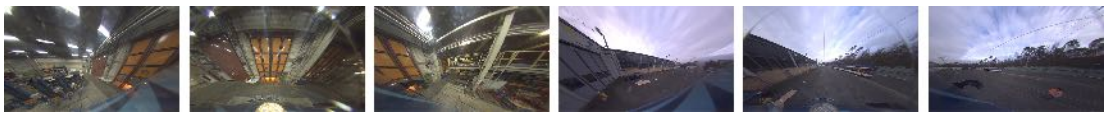
(a) Fisheye images



(b) Undistorted images using our supervisor's parameters (ROS's calibration)

(c) Undistorted images using OpenCV's calibration

(d) Further processed images with nearly parallel lines



(e) Undistorted images using our parameters

Figure 5-3: The different image sets used in the survey.

The figure shows quite well the differences among the methods. Whereas the fisheye images Fig. 5-3a have very curved lines, the undistorted images based on the intrinsic camera parameters found by a calibration process (Fig. 5-3b and Fig. 5-3c have very straight lines which eventually meet in a vanishing point. This rather strong perspective distortion stays in contrast to the fourth image set Fig. 5-3d. Here, for comparison, we have tried to correct the perspective distortion to obtain an image that looks like what is commonly seen in people's day to day life. Here, the lines in the scene which are parallel to the image plane will also be parallel in the resulting image[96]. Last but not least, there is the image set resulting from our hand tuned intrinsic camera parameters. Here, we have tried to incorporate the best of both worlds, i.e. not distorting the image perspectively too much to obtain straight lines but enough to look more natural than the fisheye image. Further, this also leads to parallel lines being not to build a vanishing point inside the image. It is evident that there is one the one hand a trade-off between having straight lines and not too stretched corners and omitting a strong perspective distortion while preserving details.

In direct comparison, you can see that for the rectified images the sense of distance in the image gets lost. It is fair to say that distances are hard to tell around the edges of

[96] Since the fisheye camera is tilted towards the ground this does not hold for the camera setup but is artificially added for a more natural view.

the fisheye camera but it works quite well in the center. In terms of image, the processed images do not lose a significant amount of information although pixels get cut off in the end. The majority of participants found the fisheye images the best and stated they could imagine teleoperating a vehicle best based on those images. The final ranking is shown in Table 5-2.

Table 5-2: The final ranking result of the survey about the best undistortion method.

rank	Image set
1	Fisheye images
2	Undistorted images using our parameters
3	Undistorted images using OpenCV's calibration
4	Further processed images with nearly parallel lines
5	Undistorted images using supervisor's parameters

These results have to be interpreted with a bit of precaution since the group is not representative for future users of the system. Neither of the participants is trained in teleoperation nor in driving a tram. Nevertheless, this gives a good first impression how the different image variants appear to humans with respect to visual information content. Interestingly, the participants agreed stronger on which is the worst approach than on which one is the best. Further, the results of this small survey confirmed our first impression about the undistorted images in Fig. 5-3b. Further, nobody voted our parameters worst. More details and complete results of this survey can be found in Chapter *Survey on image quality of processed fisheye images* in the Annex.

We have implemented ROS node[97] that synchronizes and rectifies the images. Our implementation is based on Sec. 5.3 and the C++version of the OpenCV lens correction process described in the first part of this Section. Therefore, the OpenCV class `cv::fisheye::initUndistortRectifyMap` computes undistortion and rectification maps of fisheye image, the `cv::remap` applies a generic geometrical transformation to the fisheye image using the output maps of the previous step. We provide all the parameter sets that yielded the rectified images of the survey (Table A-3).

5.6 Feature Point Detection Test

Before Stitching photos into a panorama image, it is important to find enough feature points. Observed from the development of feature point detection method, corner detection is a somehow older method and is rarely adopted in modern stitching model. In contrast to it, methods like SIFT, SURF and ORB frequently used in modern research papers. However, we still need to try codes from different methods, find feature points in one picture (Fig. 5-4) and analyse the results. Here we use the Harris Corner Detector[98] and SIFT[99] implementation of OpenCV .

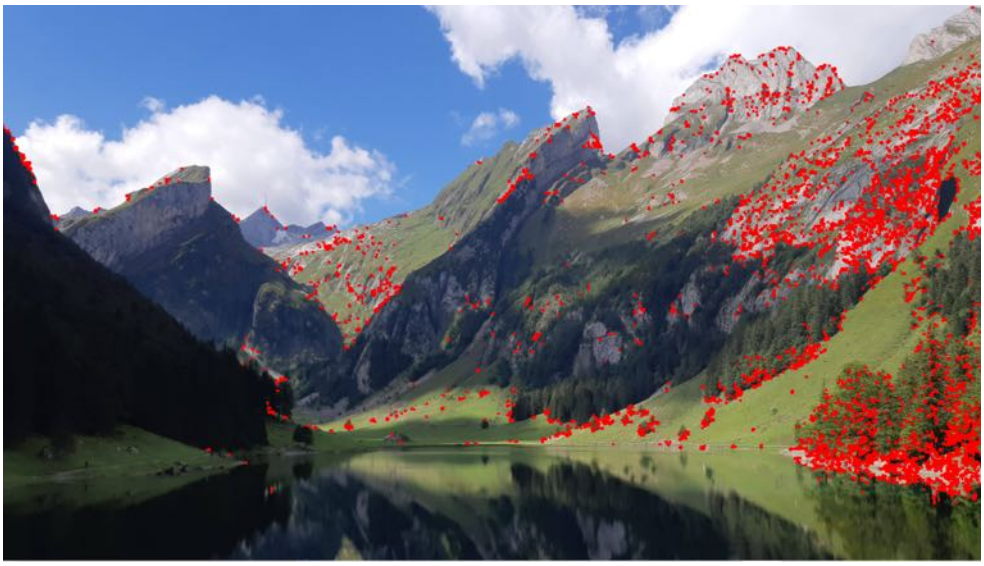
[97] The code is available in the FZD GitLab.

[98] OpenCV: Harris corner detector (2021).

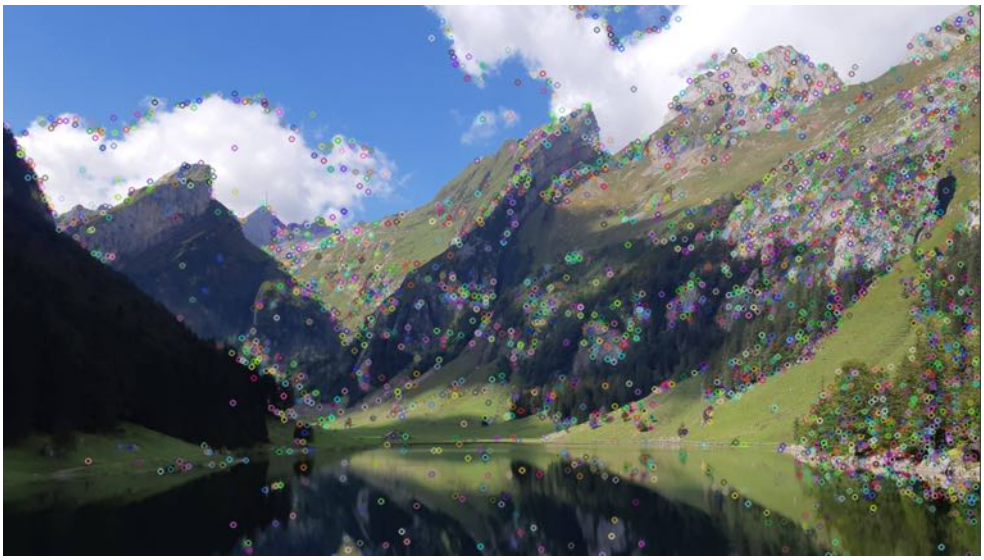
[99] OpenCV: Introduction to SIFT (Scale-Invariant Feature Transform) (2021).

Figure 5-4: Original Image.

The results (Fig. 5-5) show obvious difference between two methods. First, feature points are densely detected in SIFT. Contours in dark sides of mountains are rarely detected in Harris. Second, feature points for clouds and reflection on water almost fail in Harris. The reason is that the small contrast from cloud and sky causes difficulty to detect, and objects on the water are vague. The algorithm in Harris Detector could not handle this condition well. Experimentally we could reach the conclusion that SIFT is suitable for detecting feature points.



(a) Feature Points with Harris Corner Detector.



(b) Feature Points with SIFT.

Figure 5-5: A comparison of Harris corner detector and SIFT.

If the resolution in one image is not high enough, or contours of objects are blurred, detection methods may fail. Given that, most of methods for feature point detection are based on gradient difference, we could change the contrast and brightness in original image. For example, enhancing the contrast or reducing the brightness in some area. OpenCV gives us codes for changing parameters.

5.7 Stitching Module

After rectifying the images, a panoramic image can be created from different images using a suitable stitching approach. Following an intensive literature research, we have found a zoo of possible approaches (see Chap. 4). Based on the papers, we decided to apply the NISwGSP algorithm. It is state of the art, produces astonishing results on the test data and

beats other approaches like AANAP, AutoStitch and APAP in terms of image quality. The produced panorama images blend together nicely in the overlapping parts and are free of ghosts. As mentioned before in Chap. 4, some potential solutions would be interesting in general but out of the scope of this ADP because it would exceed the frame of this project. It was not applicable to implement algorithms from scratch which are not guaranteed to yield good results. Furthermore, deep learning based approaches that directly stitch the final image from the fisheye image would need to much data which is simply not available.

The implementation of the algorithm can be found by the authors on GitHub[100]. For adjustments to the code we made a fork[101]. In it you can find some improvements, sample files, Xcode project. The documentation of the algorithm for use is very poor, so it took some familiarization. The images to be stitched are placed in a folder in `input-42-data`. In this folder there is also a configuration file. This contains the number of images, identifies the center image and its angle. Furthermore it is defined in which order the images or which edges should be joined. As parameter of the program the folder name is given and the resulting image can be found in `result` afterwards.

5.7.1 Baseline

In order to test the functionality and reliability of the algorithm, images of the provided dataset from NISwGSP[102] and own photos were used. All images were photographed from the same position from different points of view. Figure 5-6 shows the RAW images (Fig. 5-6a and Fig. 5-6c) and the results of the algorithm (Fig. 5-6b and Fig. 5-6d).

[100] https://github.com/nothinglo/NISwGSP.

[101] https://github.com/tobka777/NISwGSP.

[102] Chen, Y.-S.; Chuang, Y.-Y.: Natural Image Stitching with the Global Similarity Prior (2016).

(a) Raw Images[102] from two perspectives – Train

(b) Panorama stitched image – Train

(c) Raw Images from three perspectives – Landscape

(d) Panorama stitched image – Landscape

Figure 5-6: Panorama stitched images from the same position with different angles.

The resulting panorama image looks like one picture. It can be determined by the fact that no dividing lines between the images or even color differences are visible. There are no blurred areas in the image (Ghosting). The image is not rectangular because image information is missing due to the different viewing angles. To get rectangular images, they still have to be cropped.

The execution time is up to 30 seconds (single core) per panoramic image depending on the input image for a resolution 800×600 Pixel. However, in our Implementation-Evaluation loop (see Fig. 1-2, recognized first for NISwGSP and later for other state of the art algorithms that those do not work optimally with the MAAS camera setup. Thus, we chose on image quality over time delay, i.e. finding a working solution without regarding the run time[103]. Although this seems like a major drawback at first glance, it is not. It is possible to reuse the expensively calculated matrices or even precalculate them

[103]This idea originated in a meeting with Mr. Pintscher and Mr. Wang of FZD.

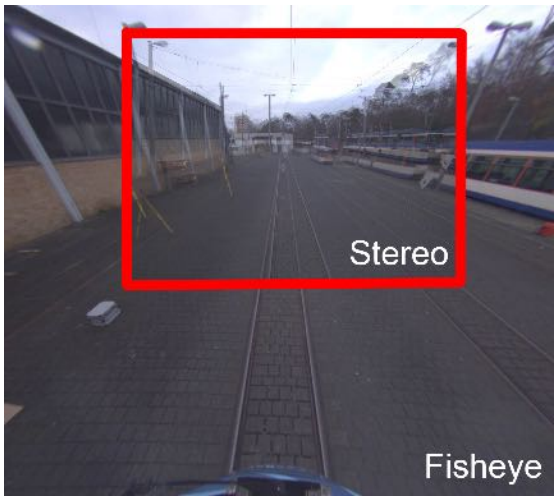
offline and then apply them online. This is also another reason, why we prioritized a high quality image. Our idea was to first get a suitable panoramic image and then perform this transformation with OpenCV.

5.7.2 First Results and Analysis

After the good results confirmed the selection for the algorithm NISwGSP, the camera images were used for stitching.

The MAAS tram has two stereo cameras and three fisheye cameras (left, middle, right). The stereo cameras overlap almost completely, so stitching these two images (Fig. 5-7a) is not purposeful.

(a) Stereocamera Stitching



(b) Stereocamera in Fisheyecamera

Figure 5-7: Difference between fisheye and stereo camera.

We decided to only stitch all three fisheye images together because together they have the largest field of view. The stereo image is just a small section of the front fisheye camera as can be seen in Fig. 5-7b.

As already shown in Section 4.4 there are two methods for calculating the MLDR. In our application the 2D method is suitable because all images in a x axle and no image in y axle are stitched to a panoramic image.

We performed first experiments with outdoor images (Fig. 5-8b). However, we got strange results as in Figure 5-9. Our analysis revealed that while feature pairs are found, they are incorrect and too few. For example, there were 71 initial feature pairs, which was reduced to 7 feature pairs with RANSAC. Figure 5-10 shows the found initial and RANSAC feature pairs.

(a) Fisheye Images from MAAS – outside

(b) Rectified Images from MAAS – outside

Figure 5-8: Images from MAAS – outside.

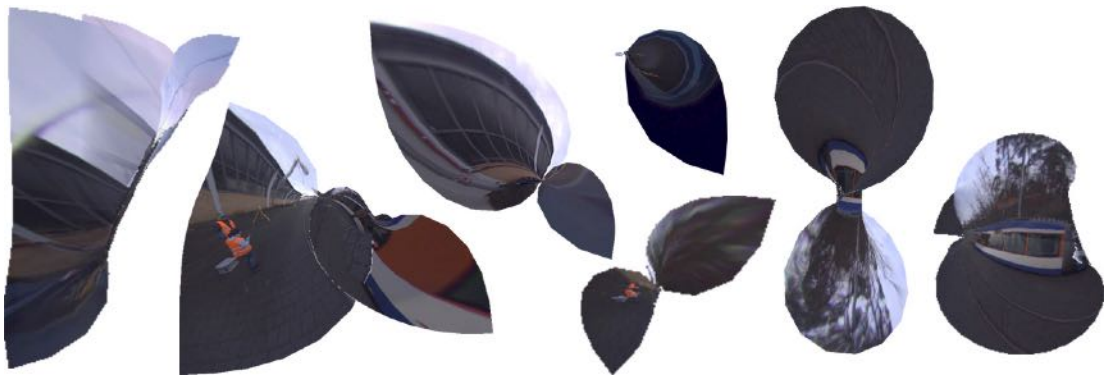
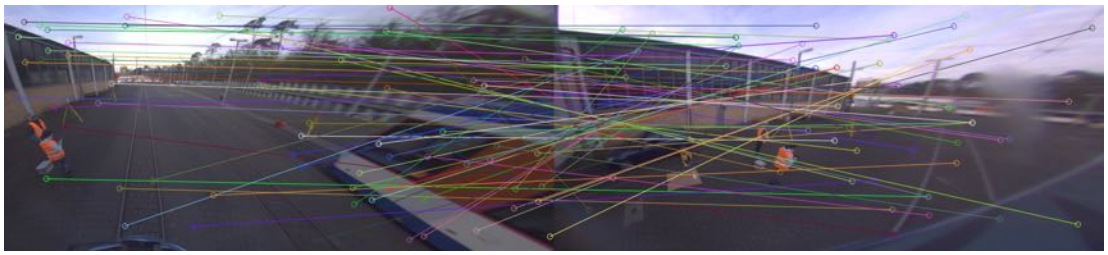


Figure 5-9: Stitching experiments.



(a) Initial feature pairs

(b) RANSAC feature pairs

Figure 5-10: Initial and RANSAC feature pairs.

In the matches, it happened that multiple feature points matched to one point, as shown by the green and yellow match in Figure 5-10b. To avoid this, we adapted the code so that this was no longer possible by using a crosscheck (similar to OpenCV's BFMatcher[104]).

However, this did not improve the quality of the features. We tested different parameters. For more initial features, we decremented `FEATURE_RATIO_TEST_THRESHOLD`[105] which kept more feature matches, but did not improve the quality of the result. Through the parameters for RANSAC like `GLOBAL_HOMOGRAPHY_MAX_INLIERS_DIST` or `GLOBAL_TRUE_PROBABIL` it is possible to adjust more iterations or the maximum distance of inliers for homography. All parameter adjustments did not lead to a visible improvement of the panoramic images, so we used the default values.

Due to the problems of the NISwGSP algorithm, an attempt was made to use other stitching approaches, which were mentioned in Chap. 4. For this purpose, the algorithms APAP[106], AANAP[107] and PtIS[108] are used. However, both algorithms produce similarly poor results (Fig. 5-11). It was also not possible to stitch the images together using the AutoStitch[109] algorithm.

[104]OpenCV: cv::BFMatcher Class Reference (2021).

[105]This corresponds to the inverse of Lowe's feature ratio r_l, whose influence is shown in Eq. (4-1),

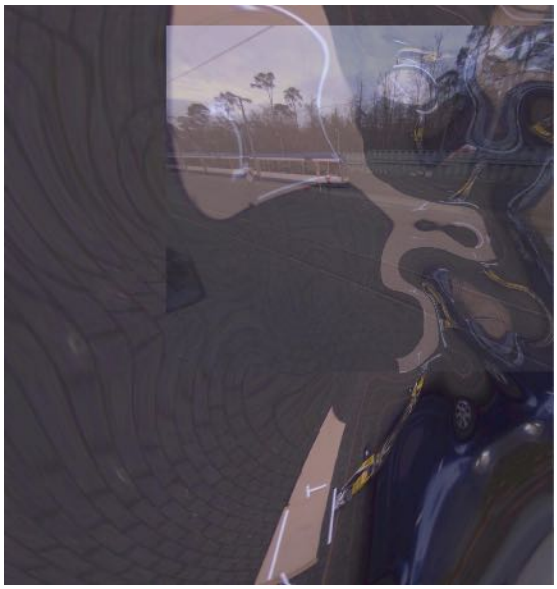
[106]MATLAB: https://cs.adelaide.edu.au/~tjchin/apap.

[107]MATLAB: https://github.com/YaqiLYU/AANAP, Python: https://github.com/lxlscut/AANAP_STITCHING.

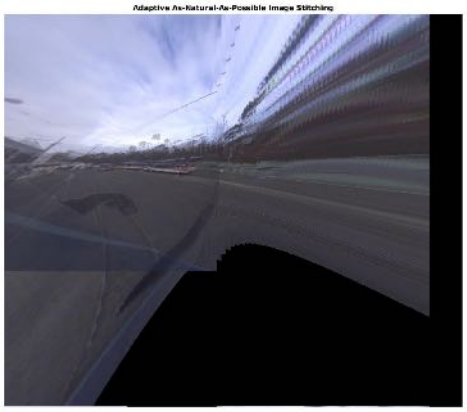
[108]https://github.com/gain2217/Robust_Elastic_Warping.

[109]Brown, M.; Lowe, D.: Automatic panoramic image stitching (2007).

(a) Image Stitching with PtIS from middle, right



(b) Image Stitching with PtIS from left, middle, right



(c) Image Stitching with AANAP from middle, right

(d) Image Stitching with APAP from left, middle

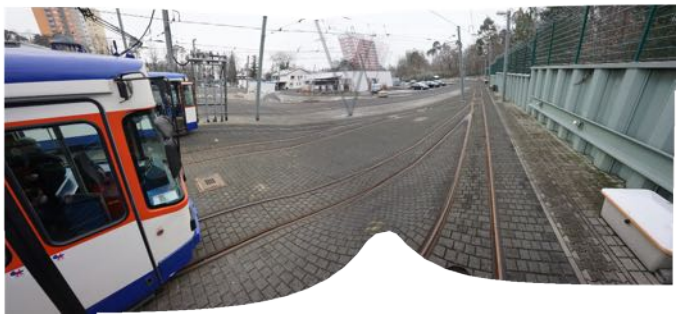
Figure 5-11: Image Stitching with AANAP and PtIS.

5.8 Experiments for Further Analysis

In the following, we explore the question of why only a few correct feature pairs were found. The fisheye cameras have a good resolution (3.17 megapixel), but the compressed 180 degree image results in a low resolution for the rectified image. Furthermore, the glass dome above the cameras creates artifacts in the overlapping parts images.

To exclude these causes, we used other images, which were provided to us by our supervisor. These were taken with a system camera (Sony α7 III, 14 mm) from the position of the fisheye camera at the tram (Fig. 5-12a).

(a) RAW images from system camera



(b) Panorama image from left and middle image

(c) Panorama image from three pictures

(d) Feature matches

(e) Mark edges for better mapping

Figure 5-12: Testing with system camera.

It turned out that it was possible to stitch the left and middle camera (Fig. 5-12b). However, there were problems when stitching the right image because wrong matches were found (Fig. 5-12d). Therefore, all three stitched images do not result in a suitable image (Fig. 5-12c). The cause may be the similar and recurring pattern of the right wall. For a test, edges of the right and middle image were marked red (Fig. 5-12e) to allow a more suitable matching to the algorithm. However, this also did not lead to a better result.

For another experiment, we wanted to take pictures from different angles and positions. To do this, we drew the floor plan of the tram to scale and marked the camera positions (Fig. 5-13c). The photos in Fig. 5-13a were taken from different positions and angles of view along the vehicle from a height of 2.10 m using a system camera (Sony α6500, 12 mm, 99° Field of View (FoV)).

(a) RAW images from system camera

(b) Panorama image

(c) Setup of tram simulation

Figure 5-13: Simulation with system camera.

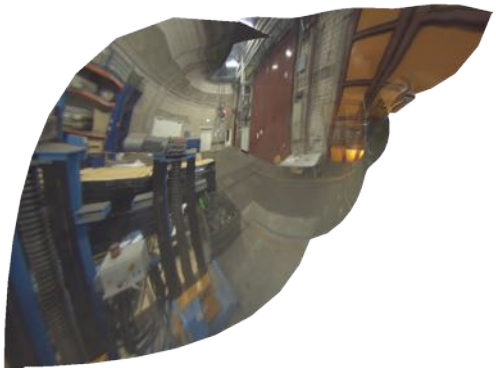
Already with the images from similar position and field of view of the fisheye cameras a good stitching is possible (Fig. 5-13b). Apart from the Field of View (FoV) distortion caused by the parallax, the image is of high quality compared to the stitched image without parallax (Fig. 5-6d).

5.9 Final Results

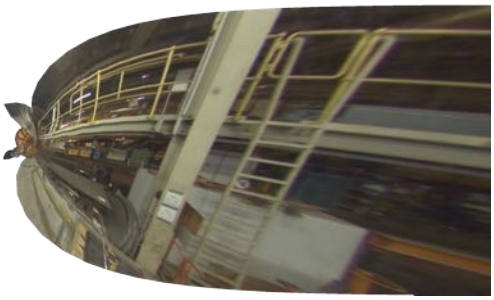
After all adjustments had been tried out, camera images were now taken from a different environment. The assumption was that there are few edges outside and that they are further away than in a hall.

(a) Rectified Images from MAAS – workshop hall

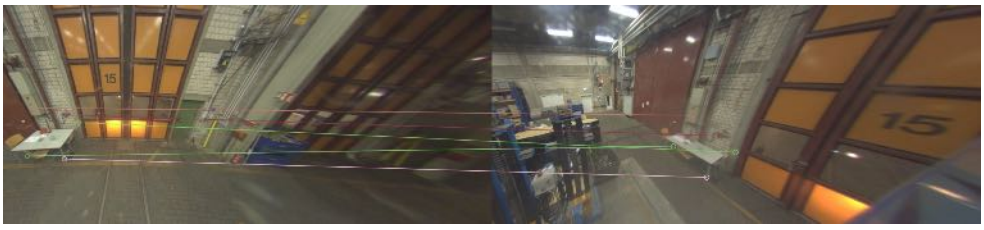
(b) Panorama image from middle and right



(c) Panorama image from left and middle



(d) Panorama image with left, middle and right images



(e) Feature matches from left and middle picture

(f) Feature matches from middle and right picture

Figure 5-14: Stitched images from MAAS - workshop hall.

Images from the workshop hall (Fig. 5-14a) were created and rectified. Fortunately, the merging of the middle and right fisheye cameras worked, as Figure 5-14b shows. From this image alone, one can already see that caused by the strong wide angle of the image a trapezoidal image with strongly tapered edges has been created. The stitched images from the middle and left cameras show the already identified problem of the fisheye camera that incorrectly matched feature pairs are selected (Fig. 5-14e). Despite the few feature matches (Initial: 33, RANSAC: 10), the stitching of the right image and middle image (Fig. 5-14f) works better than for the left and middle image, which, as already discussed, is due to the few matches found or the poor quality of the rectified image on the edges.

(a) Fisheye Images from MAAS - workshop hall

(b) Panorama image from left, middle and right

(c) Panorama image from left and middle

(d) Panorama image from middle and right

Figure 5-15: Stitched fisheye images from MAAS - workshop hall.

Furthermore, the merging of the original fisheye images was examined. The stitching works, but there are unsightly artifacts. The results (Fig. 5-15) are not rectified, and the characteristic fisheye image remains. Strong ghosting can be seen at the transitions, e.g. the water pipes.

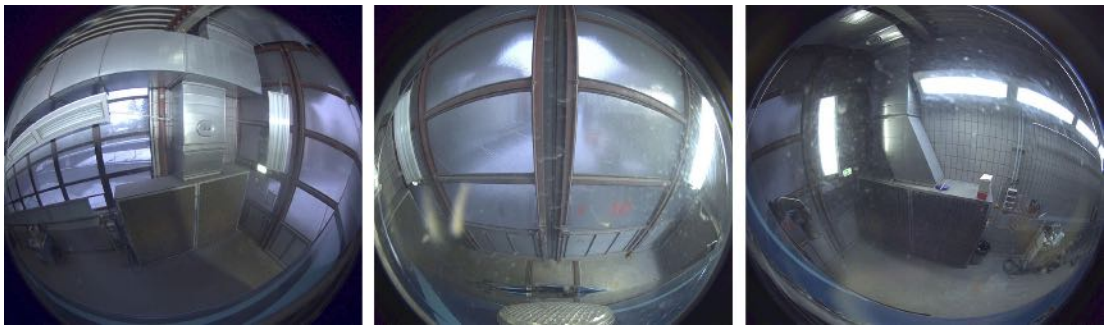


Figure 5-16: Images from MAAS - lacquering hall.

In search of more suitable image with clear edges, the camera images of the streetcar from the lacquering hall (Fig. 5-16) were used. However, no results were possible using the algorithm.

6 Evaluation of the Results

Image fusion evaluation, which is also known as image fusion quality evaluation or performance evaluation, refers to the evaluation of the actual effect of the methods used in image fusion. The purpose of the evaluation is to reflect the advantages and disadvantages of image fusion methods. Compared with the common image quality evaluation, it not only needs to evaluate the fused image, but also needs to research the relationship between the fused image and the source image. Therefore, in the construction process of image fusion quality evaluation, in addition to the quality of fusion image, the information reflection of source image in fusion image should also be considered.
The current image fusion quality evaluation methods are divided into two categories: subjective methods and objective ones. Subjective evaluation methods accord with human visual system beastly[110]. Objective evaluation methods use physical methods to measure physical characteristics of images.

6.1 Subjective Evaluation

For the performance evaluation of different image fusion methods, subjective criteria are most commonly used as the human perception of the fused image is of fundamental importance[111]. The subjective evaluation method is to evaluate the image quality directly by researchers, which is simple and intuitive. However, in the process of artificial evaluation, there will be subjective factors affecting the evaluation results, so careful experimental design is needed. So we designed a questionnaire[112] in the form of a table and about 20 respondents will participate in the questionnaire. Each respondent will be given 10 groups of images, each group of images contains the source image and the fused image. These images not only come from our different fusion algorithms, but also from the contrast images in the network. Participants don't know the source of these images and evaluate them according to different aspects of image and image fusion. Finally, we average the data of each group and classify them according to the source.
In most cases, people are users of fused images, so this method is the most reliable and direct. But the subjective evaluation method also has limitations:

- It is greatly affected by the environment. For example the display effect of the image display, the observation distance, the light and so on will affect people's judgment.
- It can only give qualitative analysis. If the difference between two images is slight, it is difficult to distinguish the quality.
- The cost of evaluation is high. It needs huge manpower and financial resources, and the efficiency is low.

[110] Zhang, X.-l. et al.: Quality Assessment of Image Fusion (2010).

[111] Blasch, E. et al.: Image quality assessment for performance evaluation (2008).

[112] The questionnaire can be found in the Annex.

6.2 Objective Evaluation

Objective evaluation method is to establish a mathematical model related to the meaning of image quality by defining some mathematical formulas. Then it calculate the evaluated image to get the digital quantity as the evaluation result. Objective evaluation method has the advantages of low cost and easy implementation, but the biggest problem of the existing objective evaluation method is that it does not fully consider the characteristics of the human visual system, so that the judgment results are often inconsistent with the subjective judgment. There are dozens of objective evaluation, which can be divided into five categories: Based on statistics, information theory, structural similarity, visual system, etc. This paper will choose the most representative evaluations in each category to explain. Finally, we will realize the algorithm of each method by computer and evaluate the results of our project objectively. In our project, we mainly use Python, OpenCV and scikit-image[113] (a.k.a. skimage) library to do the evaluation.

6.2.1 Objective Evaluation Based on Statistics

Objective Evaluation based on statistics evaluates the image quality by statistical fusion of some features of the image. This kind of method is the first one in the field of image fusion quality evaluation. Its advantage lies in its simple principle and convenient calculation. In this chapter are three methods used to evaluate the fused image: Standard deviation, Spatial frequency and the peak signal to noise ration.
Standard Deviation (STD)[114] is mainly used to measure the richness of image information. The Standard Deviation of the fused image can be expressed by the following mathematical formula.

$$\mathrm{STD} = \sqrt{\frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}\left[I(i,j)-\bar{I}\right]^2} \tag{6-1}$$

$I(i,j)$ represents the gray-scale value at pixel (i,j) and $\bar{I}$ is the mean gray-scale value of the entire image. The larger standard deviation is, the greater the difference of the gray-scale value in the fused image is. This shows that the image contains more information and are more colorful.
The Spatial Frequency (SF)[115] reflects the change rate of the gray-scale value, namely gradient. Its calculation formula is as follows:

$$\mathrm{SF} = \sqrt{\mathrm{RF}^2 + \mathrm{CF}^2} \tag{6-2}$$

[113] https://scikit-learn.org.

[114] Su-xia, X.; Tian-hua, C.: Image fusion based on regional energy and standard deviation (2010).

[115] Everson, R. M. et al.: Representation of spatial frequency and orientation (1998).

with

$$\mathrm{RF} = \sqrt{\frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}[I(i,j)-I(i,j-1)]^2}$$

$$\mathrm{CF} = \sqrt{\frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}[I(i,j)-I(i-1,j)]^2}$$

The Row Frequency (RF) represents a horizontal gradient. The Column Frequency (CF) represents the vertical gradient. It mainly evaluates the details and texture of the image, so as to reflect the clarity of the image. The bigger SF is, the clearer the image is.

The Peak Signal to Noise Ration (PSNR) is used to measure the ratio between the effective information and the noise of the image, and can reflect whether the image is distorted. It is measured in dB. The formula[116] of PSNR is shown below.

$$\mathrm{PSNR} = 10\log\frac{z^2}{\mathrm{MSE}} \tag{6-3}$$

Where z is the maximum possible pixel value of the image. Due to a very common color depth of 8 bits, it is usually 255. The bigger the PSNR index is, the higher the quality of the fused picture is.

6.2.2 Objective Evaluation Based on Information Theory

The objective evaluation based on information theory means that its theoretical basis comes from information theory. It evaluates the quality of image fusion from the perspective of information transmission. The main method used in our project is Normalized Mutual Information (NMI).
Mutual Information (MI) is used to measure the quality of information transferred from the source image to the fused image. The Mutual Information formula between two pictures is:

$$\mathrm{MI}(A,B) = \mathrm{H}(A) + \mathrm{H}(B) - \mathrm{H}(A,B) \tag{6-4}$$

The function H stands for the entropy of the image. It is mainly an objective evaluation index to measure the amount of information contained in the image:

$$\mathrm{H}(A) = -\sum_{a} P_A(a)\log p_A(a) \tag{6-5}$$

The size of entropy can be obtained directly by using the skimage Library of OpenCV. P_A represents the probability distribution of the gray-scale value of image A. We can get the

[116] Gupta, P. et al.: A modified PSNR metric based on HVS (2011).

Mutual Information (MI) between the fusion image and multiple source images by simple superposition.

$$\mathrm{MI} = \mathrm{MI}(A,F) + \mathrm{MI}(B,F) + \cdots \tag{6-6}$$

Mutual Information can directly judge the quality of the fused image F. The larger its value is, the more information the fused image gets from the source image and the more successful fusion of images is. We optimize the algorithm and standardize the Mutual Information.[117]

$$\mathrm{NMI} = n\left[\frac{MI_{AF}}{H_A + H_F} + \frac{MI_{BF}}{H_B + H_F} + \cdots\right] \tag{6-7}$$

The closer the value of the Normalized Mutual Information (NMI)[118] is to 1, the better the quality of the fused image is.

6.2.3 Objective Evaluation Based on Structural Similarity

The Evaluation based on structural similarity comes from the structural similarity theory of Wang Zhou's team at the University of Waterloo. The Structural Similarity Index Measure (SSIM) is a method to evaluate the perceived quality of digital images and videos, by comparing local patterns of pixel intensities, which have been normalized for luminance and contrast. Used to measure the similarity between two images, the SSIM index is a full reference metric, which means that a complete reference image is assumed to be known. In SSIM model, the image degradation as perceived change in structural information are taken into consideration. This means that some important perceptual phenomena should not be ignored but realized. For example, luminance masking and contrast masking terms. The luminance is the product of the illumination and the reflectance, which can be observed from the surface of an object. However, the structural information in an image is not relevant to the influence of the illumination. Therefore, it is necessary to separate these influences in the working process of algorithm by calculating the average number. The structural information is the idea that pixels have strong inter-dependencies especially when they are spatially close. Different from other techniques such as MSE or PSNR, this approach estimate relative errors. And the task is divided into three comparisons: luminance, contrast and structure.[119]

$$\mathrm{SSIM}(x,y) = \left[l(x,y)^{\alpha} \cdot c(x,y)^{\beta} \cdot s(x,y)^{\gamma}\right] \tag{6-8}$$

[117] Hossny, M.; Nahavandi, S.: Comments on 'Information measure' (2008).

[118] For better readability, we chose the substitute the arguments in parenthesis by indices.

[119] Zhou Wang et al.: Image quality assessment: from error visibility (2004).

Parameters α, β, γ here are used to show the weights of three comparisons. The measurements of three comparison measurements between samples of x and y are showed below.

$$
\begin{aligned}
l(x,y) &= \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1} \\
c(x,y) &= \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2} \\
s(x,y) &= \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3}
\end{aligned} \tag{6-9}
$$

Assuming discrete signals, μ_x is used to estimated the mean intensity of sample x. And σ_x is the standard deviation of x, which is used here to estimate signal contrast.

$$
\begin{aligned}
\mu_x &= \frac{1}{N}\sum_{i=1}^{N} x_i \\
\sigma_x &= \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)^2}
\end{aligned} \tag{6-10}
$$

In order to simplify the expression, α, β, γ can be set as 1. And $C_3 = C_2/2$. The algorithm[120] is transformed as below.

$$
\mathrm{SSIM}(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \tag{6-11}
$$

In this project we can use `skimage.measure` in Python to estimate the value of the SSIM index. The value of this index is always between -1 and 1. And the more close to 1 is, the better the quality of image is.

6.2.4 Objective Evaluation Based on Visual System

Images are evaluated in this Evaluation by Human Visual System (HVS). The Visual Information Fidelity (VIF)[121a] is a method developed by Hamid R Sheikh and Alan Bovik at University of Texas at Austin in 2006. It shows a good correlation with human judgments of visual quality and is proved to be an effective full reference image quality metric based on Natural Scene Statistics (NSS) theory. The model of this method is shown in Fig. 6-1.

[120] Wang, Z.; Simoncelli, E. P.: Multiscale structural similarity for image (2003).

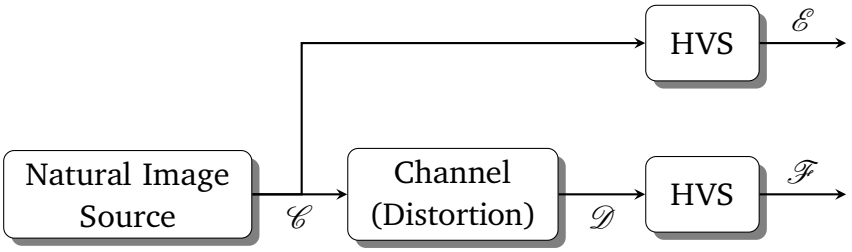


Figure 6-1: System model for VIF index.[121b]

The natural images are firstly decomposed into several sub-bands and divided into blocks. Different models are measured in the next process respectively to get the visual information by computing mutual information. Finally, the image quality value is measured by integrating visual information for all the blocks and all the sub-bands. The three models used to analyse images are the Gaussian Scale Mixture (GSM) model, the distortion model and the Human Visual System. The formulas of three models are shown below.
For Source Model: $\mathscr{C} \ni C_i = s_i U_i$
Where C_i denotes the ith random field of the reference signal in a sub-band; s_i is the ith random positive scalar; U_i is the ith Gaussian vector.
For the distortion model: $\mathscr{D} \ni D_i = g_i C_i + V_i$
Where D_i denotes the corresponding rand field in the sub-band in the test image; g_i is the scalar value and is determined by distortion Eq; V_i is a stationary additive zero mean Gaussian noise field.
For HVS model: $\mathscr{E} \ni E_i = C_i + N$, $\mathscr{F} \ni F_i = D_i + N' = g_i C_i + V_i + N'$
Where E_i and F_i denote the visual signal at the output of the HVS; N and N' are the nose of HVS. The amount of information extracted from the reference is obtained as $I(C_i, E_i)$ and the amount of information extracted from the test image is obtained as $I(C_i, E_i)$. When all sub-band information are taken into consideration, the index of the Visual Information Fidelity (VIF) can be written below [122].

$$\mathrm{VIF} = \frac{\sum_{j \in subbands} I\left(\overline{C}^{N,j}; \overline{F}^{N,j} | S^{N,j} = s^{N,j}\right)}{\sum_{j \in subbands} I\left(\overline{C}^{N,j}; \overline{E}^{N,j} | S^{N,j} = s^{N,j}\right)} \tag{6-12}$$

To evaluate the quality of image after fusion, the method Visual Information Fidelity for Fusion (VIFF) is used as performance metric. In this method, the Fusion Visual Information without Distortion Information (FVIND) for the source image I_i and fused image I_F in the bth block and kth sub-band can be defined by Equation. And also the Fusion Visual Information with Distortion Information (FVID). In conclusion, the VIFF index is shown below[123].

$$\mathrm{VIFF}_k(I_1, \ldots I_n, I_F) = \frac{\sum_b FVID_{k,b}(I_1, \ldots, I_n, I_F)}{\sum_b FVIND_{k,b}(I_1, \ldots, I_n, I_F)} \tag{6-13}$$

[121]Sheikh, H. R.; Bovik, A. C.: Image information and visual quality (2006), a: -; b: p. 4.

[122]Rezazadeh, S.; Coulombe, S.: Low-complexity computation of visual information fidelity (2010).

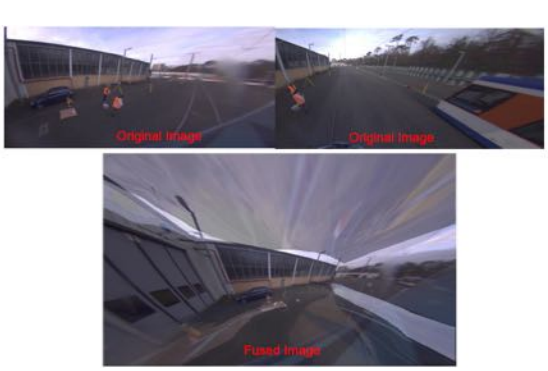
[123]Han, Y. et al.: A new image fusion performance metric (2013).

$$\mathrm{FVIND}_{k,b}(I_1,\ldots I_n,I_F) = \frac{1}{2}\log_2\left(\frac{|s^2_{k,b}C_U + \sigma^2_N I|}{|\sigma^2_N I|}\right)$$
$$\mathrm{FVID}_{k,b}(I_1,\ldots I_n,I_F) = \frac{1}{2}\log_2\left(\frac{|g^2_{k,b}s^2_{k,b}C_U + (\sigma^2_{V_{k,b}} + \sigma^2_N)I|}{|(\sigma^2_{V_{k,b}} + \sigma^2_N)I|}\right) \quad (6\text{-}14)$$

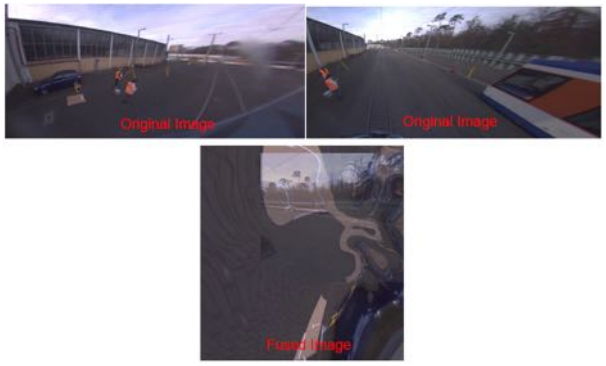
Where, the $S^2_{k,b}C_U$ means the local variance of pixels, and σ stands for the standard deviation of the source image I. The index of VIFF will grows, if the fused image shows better in visual system.

6.3 Results of the Stitched Image Evaluation

Finally, we evaluate four groups of images from three algorithms subjectively and objectively. These four groups of results are all from the algorithms we have tried and used. The details of introductions are written in previous chapters. The difference between Group C and Group D is that the original images of Group C was taken with the system camera.

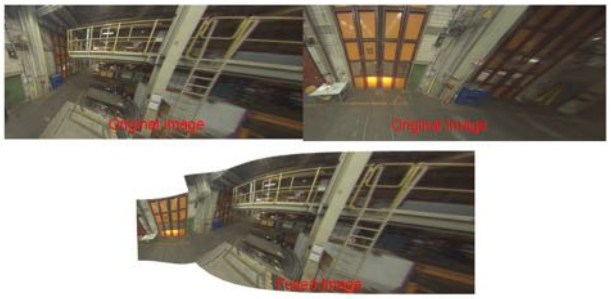


(a) Group A from AANAP



(b) Group B from PtIS

(c) Group C from NISwGSP



(d) Group D from NISwGSP

Figure 6-2: Evaluation with different groups.

The subjective and objective evaluation results of these four groups of images are shown in Table 6-1 and Table 6-2. The subjective Evaluation adopts the five point system, which represents five different grades from 1 to 5. The evaluation criteria are shown in Table 6-3.

Table 6-1: subjective evaluation results

Evaluative	Group A	Group B	Group C	Group D
Evaluation of the picture				
Visual Effect	3.25	3.88	2.5	2.63
Image Integrity	3.25	3.75	2.25	2.75
Clarity	3	3.86	2	2.5
Evaluation of the image fusion				
Seam	3	3.62	2.25	2.75
Ghost Effect	3.13	4.5	2.63	2.25
Distortion	3.38	3.88	2.38	3
Similarity to source image	3.13	4	2.38	2.75
Average Score	3.163	3.927	2.341	2.661

Table 6-2: Objective evaluation results

Picture Number Method		Group A	Group B	Group C	Group D
Evaluation based on statistics	STD	42.2	33.42	71.42	67.46
	SF	10.09	4.25	24.86	8.58
	PSNR	13.37	7.71	10.19	11.61
Evaluation based on information theory	NMI	0.21	0.16	0.18	0.11
Evaluation based on structural similarity	SSIM	0.59	0.56	0.23	0.40
Evaluation based on visual system	VIFF	0.025	0.055	0.016	0.11

Because the results of four groups are not from the same source image, they can not be compared together in the aspect of the image quality except Group A and Group B. But they can be compared in the aspect of the fusion quality. Objectively, the result of Group A from AANAP is better than that of Group B from PtIS in image information richness and clarity. And its distortion is less. In the view of the fusion quality of the four groups of images, the result of Group A is the best. Its fused image contains the highest ratio of original images' information. In addition, the similarity between its source images and the fused image is the highest. From the result of subjective evaluation, the fused image of Group A is also better than that of Group B in image quality. Most of the evaluators think that its visual effect, image integrity and clarity are better than those of Group B. Subjectively, most of the evaluators think that the result of Group C from NISwGSP is better in fusion quality. It is evaluated to have fewer seams, ghosting and distortion. Its fused image is also more similar to the source image. Due to the limitations of the objective evaluation, which computer algorithms can not accurately express our visual

feelings, we should take the results of the subjective evaluation as the main reference and the results of the objective evaluation as the auxiliary reference.

Table 6-3: Evaluation criteria

Evaluative	Grade 1	Grade 2	Grade 3	Grade 4	Grade 5
Visual Effect: represents the first subjective impression of the picture	Excellent	Good	Fair	Poor	Very Poor
Image Integrity: represents whether the image is defective or not	Totally complete	Basically complete	Fair	Some information is missing	Most of the information is missing
Clarity: represents whether the image is blurred by human eyes	Very clear	A little clear	Fair	A little blurred	Very blurred
Seam: represents the overlapping area of the two images after fusion	Almost no seam	Imperceptible distortion	Fair	Perceptible Seam	Obivous Seam
Ghost Effect: represents that the same object has repeated appearance and contour	Almost no Ghost	Imperceptible Ghost	Fair	Perceptible Ghost	Obivous Ghost
Distortion: represents whether the image is distorted (out of its shape)	Retains the original shape	Imperceptible distortion	Fair	A little distortion	Serious distortion
Similarity to Source image: represents how much information the fused image getting from the source image	Almost the same	Very similar	Fair	Not similar	Totally not the same

7 Challenges and Discussion

After practicing in the implementation chapter and the feedback of evaluation chapter, we found that the various methods we tried failed to achieve the expected results, there is no doubt that the reason behind this is worth exploring.

Firstly, we start from the project itself. Our task is to stitch the video images from several fisheye cameras and rectilinear cameras. It is worth mentioning that the resolution of fisheye images is very limited. At the same time, the contrast of the images is also very low. Therefore, we have experimented with low-resolution and high-contrast images. From the results of this experiment, a sufficient number of feature points can be found. However, the feature points that can be found on the original image of the tram fisheye camera are very insufficient for feature matching among the images. In this regard, we can not except that the contrast of the image has a great influence on the search for feature points, and we do not rule out the impact of the low resolution. Additionally, the pictures used for those advanced algorithms are taken with expensive cameras with standard zoom lenses rather that fisheye lenses and the complex scenarios included in the pictures are also well chosen.

Secondly, the important task of our ADP is to ensure the operator has a good operating experience. To do so, the final result should have a rectified image. Due to the fact that the state of the art algorithms for image stitching we not designed to work with fisheye images, it is mandatory to correct the fisheye images before they are stitched. However, image rectification will cause a further drop in image resolution, especially the area far away from center point, and a further reduction in the number of feature points. We already know that at least for the calculation of global homography we only need 4 pairs of correctly matched feature points. After checking the intermediate results of the algorithm, we found that most of the feature points matching through the RANSAC step have been filtered out. This seems to be in line with common sense, because the wrong feature point matching exists objectively. Furthermore, most of the remaining feature point matching after the RANSAC step is incorrect, and there are cases where the feature matching is concentrated in a small area which is a very tough condition for the more advanced stitching techniques which use local warping.

Last but not least, we should not forget about the limitations of homographies and the scenarios when they are applicable. For a flat scene, the camera can have rotation and translation, otherwise the camera only allows pure rotation around the image plane. Observing the layout of the tram fisheye camera, we can find that the baseline between the two cameras to the other exceeds 2 m, which will cause a very large parallax. This is effect is further magnified by the large Field of View. Although local hybrid transformation methods are more flexible, their ability to deal with parallaxes is extremely limited. Methods we have implemented like AANAP, NISwGSP could only deal with small parallax between two images, and need to be improved in the future.

To sum up, all these aspects mentioned above lead to tough challenges for us to achieve

the final goal of our ADP ideally. Actually, these problems are inherent limitation of image/video stitching in field of computer vision. Thus, all teammates and supervisors, who are conducting this ADP, as well as other experts, who are doing related researches, may need more time and attempts to explore some better solutions to overcome these difficulties.

8 Conclusion and Outlook

From this ADP, we have not only learned the principles of the teleoperation system of the MAAS project, we have also investigated and compared different fisheye camera correction and image stitching methods. Besides, we used the fisheye images as well as the images taken by ourselves to verify the effectiveness of different algorithms. Although the image stitching results are not ideal, we explored the origin behind it. In the implementation procedure, we have a deep understanding of the applicable conditions of homography and the influencing factors that affect the feature points detection and matching process. After extensively reading the relevant literature and debugging different algorithms, we have a broad understanding of commonly used image stitching methods and their respective theories. In addition, we implemented image synchronization and rectification with the help of OpenCV under the ROS framework. The architecture of ROS makes it more convenient to deploy this function on different platforms.

In the process of the project, we did not take a detour on the algorithm part. For a specific engineering problem like this, it is often impossible to find a completely suitable method from the existing literature. Due to time constraints, we cannot try more image stitching algorithms. If we have the opportunity in the future or others are interested in continuing the project, we would suggest to change the layout of the tram camera so that the optical axis of the cameras are as close as possible. In this way, the severe distortion and ghosting effects caused by parallax could be improved. Further, we suggest using no fisheye cameras, but compensate the loss of FoV with one or two additional cameras based on the focal length of the lenses. This will help to better match the detected feature points in the different images. In fact, image stitching with large parallaxes is currently a problem that needs to be solved urgently in the academic world. The publication of many methods has brought new approaches, but the problem cannot be solved perfectly. Although the 3D Reconstruction method is computationally intensive and time consuming, it is theoretically feasible and it is worth trying as an potential solution.

In the future, more further optimized algorithms are expected to be proposed, which can be applied in more complicated situations, such as to stitch images with low-quality or captured by cameras located with wide baseline. Despite those results, we have also shown that the majority of people that participated in our survey would consider the unprocessed fisheye images sufficient to teleoperate a tram. Needless to say, this needs further investigation but it opens up the question whether image rectification and stitching to a panoramic view is necessary at all.

Annex

Subjective Evaluation Questionnaire

Questionnaire about evaluation of the image fusion

Please rate the following four groups of images subjectively. Each group contains two original images and a fused image.

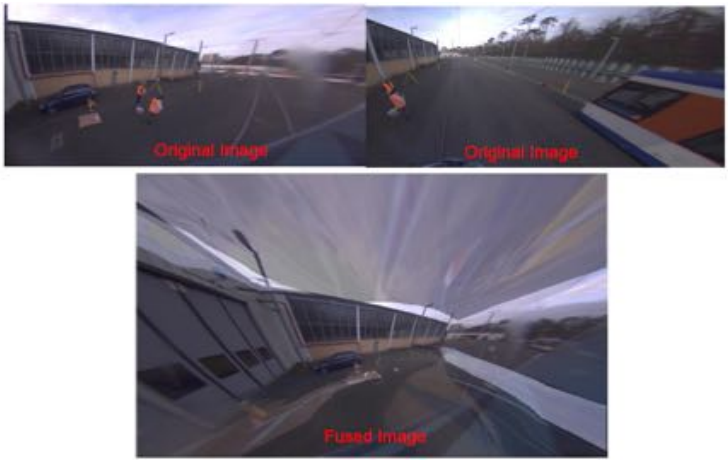


Figure A-1: Questionnaire Group A

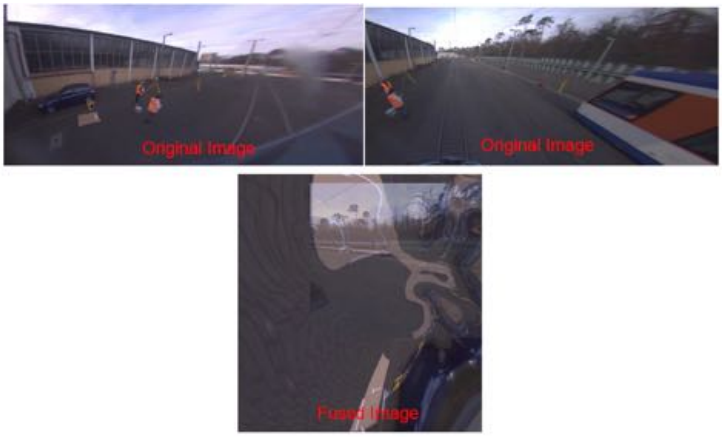


Figure A-2: Questionnaire Group B

Figure A-3: Questionnaire Group C

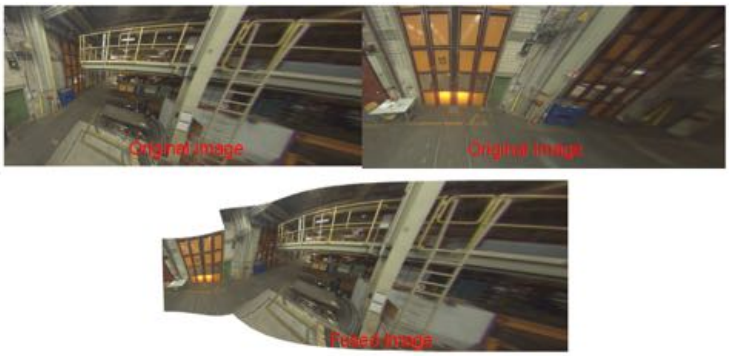


Figure A-4: Questionnaire Group D

Table A-1: subjective evaluation results

Evaluative	Group A	Group B	Group C	Group D
Evaluation of the picture				
Visual Effect				
Image Integrity				
Clarity				
Evaluation of the image fusion				
Seam				
Ghost Effect				
Distortion				
Similarity to source image				
Average Score				
*Subjective Evaluation adopts the five point system, which represents five different grades from 1 to 5.				

Table A-2: Evaluation criteria

Evaluative	Grade 1	Grade 2	Grade 3	Grade 4	Grade 5
Visual Effect: represents the first subjective impression of the picture	Excellent	Good	Fair	Poor	Very Poor
Image Integrity: represents whether the image is defective or not	Totally complete	Basically complete	Fair	Some information is missing	Most of the information is missing
Clarity: represents whether the image is blurred by human eyes	Very clear	A little clear	Fair	A little blurred	Very blurred
Seam: represents the overlapping area of the two images after fusion	Almost no seam	Imperceptible distortion	Fair	Perceptible Seam	Obivous Seam
Ghost Effect: represents that the same object has repeated appearance and contour	Almost no Ghost	Imperceptible Ghost	Fair	Perceptible Ghost	Obivous Ghost
Distortion: represents whether the image is distorted (out of its shape)	Retains the original shape	Imperceptible distortion	Fair	A little distortion	Serious distortion
Similarity to Source image: represents how much information the fused image getting from the source image	Almost the same	Very similar	Fair	Not similar	Totally not the same

Survey on image quality of processed fisheye images

As we have already briefly explained in Sec. 5.5, we wanted to achieve results of high quality. In order to achieve this, we have collected a variety of methods to evaluate our final results, which are discussed in Chap. 6. These were meant to evaluate the stitched video stream. Earlier in this work, we discussed in detail that unfortunately, the conditions of the problem were far suboptimal and lead to unsatisfactory results with the image stitching. Nonetheless, we wanted to unsure that at least our undistorted images are satisfactory and sufficient for a human operator to teleoperate a tram. To accomplish a quantified measure for image quality and usability in teleoperation task, we have created another survey on the three processed fisheye images. We have asked 55 to rank the image series found in Fig. 5-3[124]. The undistortion parameters for Ours, ROS and OpenCV are listed in Table A-3. The rectification matrix **R** is always the identity matrix in all cases. The focus was set on subjective image quality and which series would be most suitable for them to teleoperate a vehicle (tram). As far as we know, none of the participants was given training in teleoperation or has ever driven a tram, so the results have to be interpreted with care. The final results are shown in Fig. A-5. To our surprise, the majority of participants selected the raw fisheye images best. The parameters we have found have been voted second which indicates that it was worth the effort. The parameters, we have received from the camera calibration process of OpenCV and the "Parallel" images[125] have been chose third and forth. The images that have been undistorted using the parameters that our supervisors have produces through the ROS camera calibration process have been voted the least good. The participants agreed even more strongly on this rank (80 %) than on any of the other ranks. This confirmed our initial impression about these parameters.

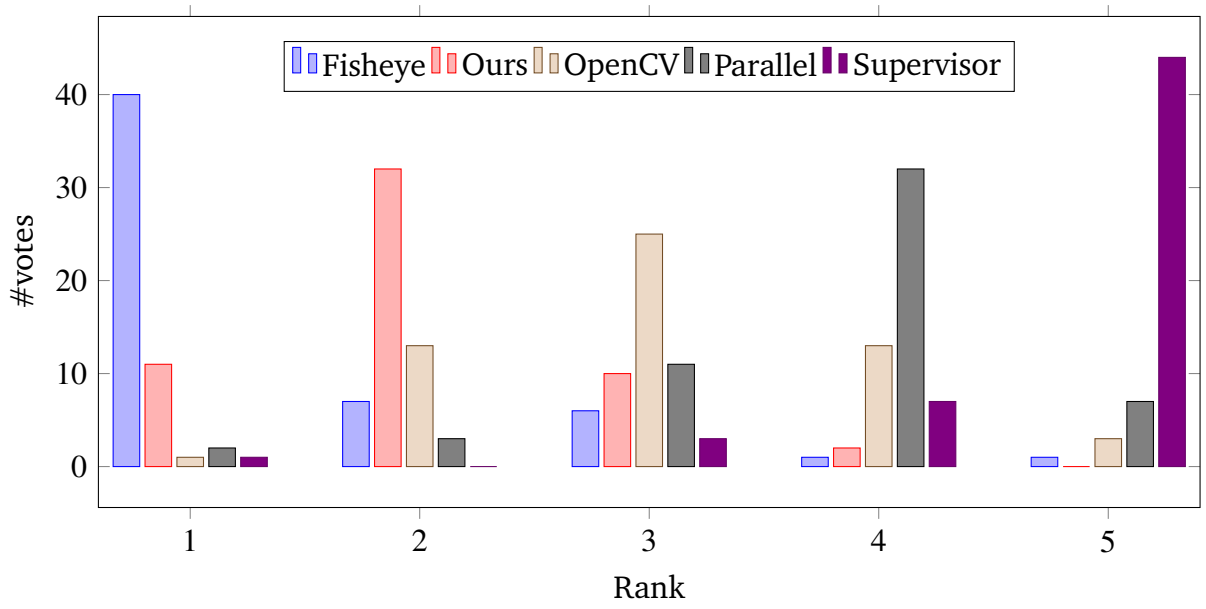


Figure A-5: The result of our survey on image quality for different undistortion methods.

[124]We did not state which one is our approach in order not to bias participants form our environment.

[125]The images have been perspectively transformed from the undistorted ones.

Table A-3: Camera matrix and distortion coefficients used in the survey.

camera	parameter	ROS	OpenCV	Ours
left	**K**	$\begin{pmatrix} 613.529 & -0.988 & 818.831 \\ 0.0 & 610.583 & 770.749 \\ 0.0 & 0.0 & 1.0 \end{pmatrix}$	$\begin{pmatrix} 726.122 & 0.0 & 790.367 \\ 0.0 & 727.801 & 752.4402 \\ 0.0 & 0.0 & 1.0 \end{pmatrix}$	$\begin{pmatrix} 580.0 & 0.0 & 818.0 \\ 0.0 & 580.0 & 786.0 \\ 0.0 & 0.0 & 1.0 \end{pmatrix}$
	d	$\begin{pmatrix} -0.06816418906630581 \\ 0.01735634023150791 \\ -0.01196330551732261 \\ 0.002955506433608474 \end{pmatrix}^T$	$\begin{pmatrix} -0.29917768 \\ 0.07276122 \\ 0.00537864 \\ 0.0039654 \end{pmatrix}^T$	$\begin{pmatrix} 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \end{pmatrix}^T$
middle	**K**	$\begin{pmatrix} 516.935 & 0.0 & 811.5 \\ 0.0 & 516.935 & 724.5 \\ 0.0 & 0.0 & 1.0 \end{pmatrix}$	$\begin{pmatrix} 840.038 & 0.0 & 804.004 \\ 0.0 & 859.818 & 781.540 \\ 0.0 & 0.0 & 1.0 \end{pmatrix}$	$\begin{pmatrix} 580.0 & 0.0 & 808.5 \\ 0.0 & 572.9350 & 805.5 \\ 0.0 & 0.0 & 1.0 \end{pmatrix}$
	d	$\begin{pmatrix} 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \end{pmatrix}^T$	$\begin{pmatrix} -0.37169199 \\ 0.11896759 \\ 0.006112 \\ 0.00124474 \end{pmatrix}^T$	$\begin{pmatrix} 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \end{pmatrix}^T$
right	**K**	$\begin{pmatrix} 615.072 & 1.195 & 802.080 \\ 0.0 & 613.216 & 771.158 \\ 0.0 & 0.0 & 1.0 \end{pmatrix}$	$\begin{pmatrix} 980.946 & 0.0 & 809.527 \\ 0.0 & 975.870 & 759.407 \\ 0.0 & 0.0 & 1.0 \end{pmatrix}$	$\begin{pmatrix} 580.0 & 0.0 & 802.0 \\ 0.0 & 580.0 & 789.0 \\ 0.0 & 0.0 & 1.0 \end{pmatrix}$
	d	$\begin{pmatrix} -0.07394001050687238 \\ 0.01452297219933673 \\ -0.005031993599562873 \\ 0.0003782017752643325 \end{pmatrix}^T$	$\begin{pmatrix} -0.32048808 \\ -0.00953996 \\ -0.00168821 \\ 0.00072801 \end{pmatrix}^T$	$\begin{pmatrix} 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \end{pmatrix}^T$

Bibliography

AG, N.: NB3800 MediaRail (2019)
AG, NetModule: NB3800 MediaRail, 2019

Arts Management Systems: Time on Computer Differs From Server (2021)
Arts Management Systems: Time on Computer Differs From Server | Arts Management Systems, URL: https://help.theatremanager.com/theatre-manager-online-help/time-computer-differs-server, 2021, visited on 02/21/2021

Bay, H. et al.: SURF: Speeded Up Robust Features (2006)
Bay, Herbert; Tuytelaars, Tinne,; Van Gool, Luc: "SURF: Speeded Up Robust Features", in: *Computer Vision – ECCV 2006*, pp. 404–417, 2006

Blasch, E. et al.: Image quality assessment for performance evaluation (2008)
Blasch, Erik; Li, Xiaokun; Chen, Genshe,; Li, Wenhua: "Image quality assessment for performance evaluation of image fusion", in: *2008 11th International Conference on Information Fusion*, IEEE, pp. 1–6, 2008

Brown, M. et al.: Automatic panoramic image stitching (2007)
Brown, Matthew; Lowe, David G: Automatic Panoramic Image Stitching using Invariant Features, in: International Journal of Computer Vision, Vol. 74, pp. 59–73, 2007

Chen, K. et al.: Vanishing Point Guided Natural Image Stitching (2020)
Chen, Kai; Yao, Jian; Tu, Jingmin; Liu, Yahui; Li, Yinxuan,; Li, Li: Vanishing Point Guided Natural Image Stitching, 2020

Chen, Y.-S. et al.: Natural Image Stitching with the Global Similarity Prior (2016)
Chen, Yu-Sheng; Chuang, Yung-Yu: "Natural Image Stitching with the Global Similarity Prior", in: *Computer Vision – ECCV 2016*, pp. 186–201, 2016

Edmund Optics GmbH: 5mm Brennweite, Weitwinkelobjektiv mit geringer Verzeichnung (2021)
Edmund Optics GmbH: 5mm Brennweite, Weitwinkelobjektiv mit geringer Verzeichnung, URL: https://www.edmundoptics.de/p/5mm-fl-wide-angle-low-distortion-lens/23289, 2021, visited on 02/17/2021

Elsevier B.V.: Haar Function (2021)
Elsevier B.V.: Haar Function, URL: https://www.sciencedirect.com/topics/engineering/haar-function, 2021, visited on 02/21/2021

Endsley, M. R.: Situation awareness global assessment technique (SAGAT) (1988)
Endsley, Mica R.: Situation awareness global assessment technique (SAGAT), in: Proceedings of the IEEE 1988 National Aerospace and Electronics Conference, 789–795 vol.3, 1988

Everson, R. M. et al.: Representation of spatial frequency and orientation (1998)
Everson, R. M.; Prashanth, A. K.; Gabbay, M.; Knight, B. W.; Sirovich, L.,; Kaplan, E.: Representation of spatial frequency and orientation in the visual cortex, in: Proceedings of the National Academy of Sciences, Vol. 95, pp. 8334–8338, 1998

Flynn, J. et al.: DeepStereo: learning to predict new views (2016)
Flynn, John; Neulander, Ivan; Philbin, James,; Snavely, Noah: "DeepStereo: Learning to Predict New Views From the World's Imagery", in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016

García, J.: 3D Reconstruction for Optimal Representation (2015)
García, José: 3D Reconstruction for Optimal Representation of Surroundings in Automotive HMIs, Based on Fisheye Multi-Camera Systems, 2015

Georg, J. M. et al.: Teleoperated Driving (2018)
Georg, Jean Michael; Feiler, Johannes; DIermeyer, Frank,; Lienkamp, Markus: Teleoperated Driving, a Key Technology for Automated Driving? Comparison of Actual Test Drives with a Head Mounted Display and Conventional Monitors∗, in: IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC, pp. 3403–3408, 2018

GmbH, H. mobilo: Über 125 Jahre Nahverkehr in Darmstadt (2021)
GmbH, HEAG mobilo: URL: https://www.heagmobilo.de/de/historie, 2021, visited on 02/22/2021

GmbH, H. mobilo: Die Straßenbahnen der HEAG mobilo (2021)
GmbH, HEAG mobilo: Die Straßenbahnen der HEAG mobilo, URL: https://www.heagmobilo.de/de/strassenbahnen-der-heag-mobitram, 2021, visited on 02/22/2021

Gupta, P. et al.: A modified PSNR metric based on HVS (2011)
Gupta, P.; Srivastava, P.; Bhardwaj, S.,; Bhateja, V.: "A modified PSNR metric based on HVS for quality assessment of color images", in: *2011 International Conference on Communication and Industrial Application*, pp. 1–4, 2011

Han, Y. et al.: A new image fusion performance metric (2013)
Han, Yu; Cai, Yunze; Cao, Yin,; Xu, Xiaoming: A new image fusion performance metric based on visual information fidelity, in: Information Fusion, Vol. 14, pp. 127–135, 2013

He, B. et al.: Parallax-Robust Surveillance Video Stitching (2015)
He, Botao; Yu, Shaohua: Parallax-Robust Surveillance Video Stitching, in: Sensors, Vol. 16, p. 7, 2015

Henry, M.: Video Compression Explained (2010)
Henry, Mike: Video Compression Explained, URL: https://wordpress.lensrentals.com/blog/2010/01/video-compression-explained, 2010, visited on 02/21/2021

Hossny, M. et al.: Comments on 'Information measure' (2008)
Hossny, Mo; Nahavandi, Saeid: Comments on 'Information measure for performance of image fusion', in: Electronics Letters, Vol. 44, pp. 1066–1067, 2008

Hou, W. et al.: Digital deformation model for fisheye image. (2012)
Hou, Wenguang; Ding, Mingyue; Qin, Nannan,; Lai, Xudong: Digital deformation model for fisheye image rectification, in: Optics Express, Vol. 20, pp. 22252–22261, 2012

IDS: UI-5270CP Rev. 2 (2021)
IDS: UI-5270CP Rev. 2, URL: https://de.ids-imaging.com/store/ui-5270cp-rev-2.html, 2021, visited on 02/17/2021

IDS: UI-5270FA (2021)
IDS: UI-5270FA, URL: https://de.ids-imaging.com/store/ui-5270fa.html, 2021, visited on 02/17/2021

Jiang, K.: Calibrate fisheye lens using OpenCV – part 1 (2017)
Jiang, Kenneth: Calibrate fisheye lens using OpenCV – part 1, URL: https://medium.com/@kennethjiang/calibrate-fisheye-lens-using-opencv-333b05afa0b0, 2017, visited on 12/21/2021

Jiang, K.: Calibrate fisheye lens using OpenCV – part 2 (2017)
Jiang, Kenneth: Calibrate fisheye lens using OpenCV – part 2, URL: https://medium.com/@kennethjiang/calibrate-fisheye-lens-using-opencv-part-2-13990f1b157f, 2017, visited on 12/21/2021

Jiaya Jia et al.: Eliminating structure and intensity misalignment (2005)
Jiaya Jia; Chi-Keung Tang: "Eliminating structure and intensity misalignment in image stitching", in: *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, vol. 2, 1651–1658 Vol. 2, 2005

Kaehler, A.; Bradski, G.: Learning OpenCV 3 (2016)
Kaehler, Adrian; Bradski, Gary: Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library, 1st. Auflage, O'Reilly Media, Inc., 2016

Kannala, J. et al.: A generic camera model and calibration method (2006)
Kannala, J.; Brandt, S. S.: A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses, in: IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 28, pp. 1335–1340, 2006

Kim, J.: Panoramic Image Communication for Mobile Application (2017)
Kim, Jaejoon: Panoramic Image Communication for Mobile Application using Content-Aware Image Resizing Method, in: International Journal on Advanced Science, Engineering and Information Technology, Vol. 7, p. 338, 2017

Lensation GmbH: Lensagon BF10M19828S118 – Lensation GmbH (2021)
Lensation GmbH: Lensagon BF10M19828S118 – Lensation GmbH, URL: https://www.lensation.de/product/BF10M19828S118, 2021, visited on 02/17/2021

Levin, A. et al.: Seamless Image Stitching in the Gradient Domain (2004)
Levin, Anat; Zomet, Assaf; Peleg, Shmuel,; Weiss, Yair: "Seamless Image Stitching in the Gradient Domain", in: *Computer Vision - ECCV 2004*, pp. 377–389, 2004

Li, A. et al.: Image Stitching Based on Planar Region Consensus (2020)
Li, Aocheng; Guo, Jie,; Guo, Yanwen: Image Stitching Based on Planar Region Consensus, 2020

Lichiardopol, S.: A Survey on Teleoperation (2007)
Lichiardopol, S: A survey on teleoperation, in: DCT rapporten, Vol. 2007.155, 2007

Lichvar, M.: Combining PTP with NTP to Get the Best of Both Worlds (2016)
Lichvar, Miroslav: Combining PTP with NTP to Get the Best of Both Worlds, URL: https://www.redhat.com/en/blog/combining-ptp-ntp-get-best-both-worlds, 2016, visited on 02/21/2021

Lin, C. et al.: Adaptive as-natural-as-possible image stitching (2015)
Lin, C.; Pankanti, S. U.; Ramamurthy, K. N.,; Aravkin, A. Y.: “Adaptive as-natural-as-possible image stitching”, in: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1155–1163, 2015

Lin, w.-y. et al.: Smoothly varying affine stitching (2011)
Lin, wen-yan; Liu, Siying; Matsushita, Yasuyuki; Ng, Tian,; Cheong, Loong: “Smoothly varying affine stitching”, in: pp. 345–352, 2011

Lindeberg, T.: Scale Invariant Feature Transform (2012)
Lindeberg, T.: Scale Invariant Feature Transform, in: Scholarpedia, Vol. 7, p. 10491, 2012

Liu, Y. et al.: Critical Assessment of Correction Methods (2016)
Liu, Y.; Tian, C.,; Huang, Y.: Critical Assessment of Correction Methods for Fisheye Lens Distortion, 2016

Liu, Y. et al.: Fisheye image distortion correction. (2020)
Liu, Y.; Zhang, B.; Liu, N.; Li, H.,; Zhu, J.: “Fisheye Image Distortion Correction Based on Spherical Perspective Projection Constraint”, in: *2020 IEEE International Conference on Mechatronics and Automation (ICMA)*, 2020

Lowe, D. G.: Distinctive image features (2004)
Lowe, David G.: Distinctive Image Features from Scale-Invariant Keypoints, in: International Journal of Computer Vision, Vol. 60, pp. 91–110, 2004

Luckas, V.: Mensch-Maschine-Kommunikation – Wahrnehmung (2015)
Luckas, Volker: Mensch-Maschine-Kommunikation – Wahrnehmung, University Lecture, 2015

Lyu, W. et al.: A survey on image and video stitching (2019)
Lyu, Wei; Zhou, Zhong; Chen, Lang,; Zhou, Yi: A survey on image and video stitching. Virtual Reality & Intelligent Hardware, in: Virtual Reality & Intelligent Hardware, Vol. 1, pp. 55–83, 2019

Mittal, S. et al.: A Survey Of Architectural Approaches for Data Compression (2016)
Mittal, S.; Vetter, J. S.: A Survey Of Architectural Approaches for Data Compression in Cache and Main Memory Systems, in: IEEE Transactions on Parallel and Distributed Systems, Vol. 27, pp. 1524–1536, 2016

Mordvintsev, A. et al.: Feature Matching – OpenCV-Python Tutorials 1 documentation (2013)
Mordvintsev, Alexander; Revision, Abid K.: Feature Matching – OpenCV-Python Tutorials 1 documentation, URL: https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_feature2d/py_matcher/py_matcher.html, 2013, visited on 02/21/2021

Neumeier, S. et al.: Teleoperation: The Holy Grail to Solve Problems of Automated Driving? (2019)
Neumeier, Stefan; Wintersberger, Philipp; Frison, Anna Katharina; Becher, Armin; Facchi, Christian,; Riener, Andreas: “Teleoperation: The Holy Grail to Solve Problems of Automated Driving? Sure, but Latency Matters”, in: pp. 186–197, 2019

Niemeyer, G. et al.: Telerobotics (2016)
Niemeyer, Günter; Preusche, Carsten; Stramigioli, Stefano,; Lee, Dongjun: Telerobotics, in: Siciliano, Bruno; Khatib, Oussama (Hrsg.): Springer Handbook of Robotics, Springer International Publishing, 2016

Open Source Robotics Foundation, Inc.: message_filters/ApproximateTime (2010)
Open Source Robotics Foundation, Inc.: message_filters/ApproximateTime, URL: http://wiki.ros.org/message_filters/ApproximateTime, 2010, visited on 02/17/2021

Open Source Robotics Foundation, Inc.: ROS/Concepts (2014)
Open Source Robotics Foundation, Inc.: ROS/Concepts, URL: http://wiki.ros.org/ROS/Concepts, 2014, visited on 02/18/2021

Open Source Robotics Foundation, Inc.: ROS/Introduction (2018)
Open Source Robotics Foundation, Inc.: ROS/Introduction, URL: http://wiki.ros.org/ROS/Introduction, 2018, visited on 02/18/2021

OpenCV: Basic concepts of the homography (2021)
OpenCV: Basic concepts of the homography, URL: https://docs.opencv.org/master/d9/dab/tutorial_homography.html, 2021, visited on 02/17/2021

OpenCV: Camera Calibration and 3D Reconstruction (2021)
OpenCV: Camera Calibration and 3D Reconstruction, URL: https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html, 2021, visited on 02/17/2021

OpenCV: cv::BFMatcher Class Reference (2021)
OpenCV: cv::BFMatcher Class Reference, URL: https://docs.opencv.org/3.4.13/d3/da1/classcv_1_1BFMatcher.html, 2021, visited on 02/22/2021

OpenCV: Fisheye camera model (2021)
OpenCV: Fisheye camera model, URL: https://docs.opencv.org/3.4/db/d58/group__calib3d__fisheye.html, 2021, visited on 02/17/2021

OpenCV: Harris corner detector (2021)
OpenCV: Harris corner detector, URL: https://docs.opencv.org/3.4/d4/d7d/tutorial_harris_detector.html, 2021, visited on 02/22/2021

OpenCV: Introduction to SIFT (Scale-Invariant Feature Transform) (2021)
OpenCV: Introduction to SIFT (Scale-Invariant Feature Transform), URL: https://docs.opencv.org/master/da/df5/tutorial_py_sift_intro.html, 2021, visited on 02/22/2021

Peleg, S. et al.: Mosaicing on adaptive manifolds (2000)
Peleg, S.; Rousso, B.; Rav-Acha, A.,; Zomet, A.: Mosaicing on adaptive manifolds, in: IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, pp. 1144–1154, 2000

Pintscher, P.: Teleoperated Driving (2020)
Pintscher, Patrick: Teleoperated Driving, 2020

Pintscher, P. et al.: Feasibility Analysis for Automation (2021)
Pintscher, Patrick; Ruppert, Timm: Feasibility Analysis for Automation and Assistance Systems of Trams, URL: https://www.fzd.tu-darmstadt.de/forschung/research_projects_fzd/maas_fzd/index.en.jsp, 2021, visited on 01/11/2021

Rezazadeh, S. et al.: Low-complexity computation of visual information fidelity (2010)
Rezazadeh, S.; Coulombe, S.: “Low-complexity computation of visual information fidelity in the discrete wavelet domain”, in: *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2438–2441, 2010

Ross, K. W. et al.: Delay and Loss in Packet-Switched Networks (2000)
Ross, Keith W.; Kurose, James F.: URL: https://www.net.t-labs.tu-berlin.de/teaching/computer_networking/01.06.htm, 2000, visited on 02/22/2021

Samsung Electronics GmbH: datenblatt LC49RG94SSUXZG (2019)
Samsung Electronics GmbH: Samsung Monitor LC49RG94SSUXZG, URL: https://images.samsung.com/is/content/samsung/p5/de/display/pdf/Datenblatt_C49RG94SSU.pdf, 2019, visited on 02/18/2021

Sheikh, H. R. et al.: Image information and visual quality (2006)
Sheikh, H. R.; Bovik, A. C.: Image information and visual quality, in: IEEE Transactions on Image Processing, Vol. 15, pp. 430–444, 2006

Szeliski, R.: Computer vision: algorithms and applications (2010)
Szeliski, Richard: Computer vision: algorithms and applications, Springer Science & Business Media, 2010

Szeliski, R.: Image alignment and stitching (2007)
Szeliski, Richard: Image Alignment and Stitching: A Tutorial, in: Foundations and Trends® in Computer Graphics and Vision, Vol. 2, pp. 1–104, 2007

Ufer, N. et al.: Deep Semantic Feature Matching (2017)
Ufer, Nikolai; Ommer, Bjorn: “Deep Semantic Feature Matching”, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017

Uyttendaele, M. et al.: Eliminating ghosting and exposure artifacts (2001)
Uyttendaele, M.; Eden, A.,; Skeliski, R.: “Eliminating ghosting and exposure artifacts in image mosaics”, in: *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 2, pp. II–II, 2001

Wang, Z. et al.: Multiscale structural similarity for image (2003)
Wang, Zhou; Simoncelli, Eero P: “Multiscale structural similarity for image quality assessment”, in: *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, IEEE, vol. 2, pp. 1398–1402, 2003

Wikipedia contributors: Homography (computer vision) (2020)
Wikipedia contributors: Homography (computer vision), URL: https://en.wikipedia.org/wiki/Homography_(computer_vision), 2020, visited on 02/17/2021

Wikipedia contributors: Parallax (2021)
Wikipedia contributors: Parallax, URL: https://en.wikipedia.org/wiki/Parallax, 2021, visited on 02/17/2021

Winner, H.: Handbuch Fahrassistenzsysteme (2015)
Winner, Hermann: Handbuch Fahrassistenzsysteme, in: Winner, Hermann; Hakuli, Stephan; Lotz, Felix,; Singer, Christina (Hrsg.): Handbuch Fahrerassistenzsysteme, Springer Vieweg, Wiesbaden, 2015

Su-xia, X. et al.: Image fusion based on regional energy and standard deviation (2010)
Su-xia, X.; Tian-hua, C.: "Image fusion based on regional energy and standard deviation", in: *2010 2nd International Conference on Signal Processing Systems*, vol. 1, pp. V1-739-V1–743, 2010

Zaragoza, J. et al.: As-projective-as-possible image stitching with Moving DLT (2013)
Zaragoza, J.; Chin, T.-J.; Brown, M.,; Suter, D.: "As-projective-as-possible image stitching with Moving DLT", in: *In Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2013

Zhang, F. et al.: Parallax-Tolerant Image Stitching (2014)
Zhang, F.; Liu, F.: "Parallax-Tolerant Image Stitching", in: *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3262–3269, 2014

Zhang, X.-l. et al.: Quality Assessment of Image Fusion (2010)
Zhang, Xiao-lin; Liu, Zhi-fang; Kou, Yong; Dai, Jin-bo,; Cheng, Zhi-meng: "Quality Assessment of Image Fusion Based on Image Content and Structural Similarity", in: *2010 2nd International Conference on Information Engineering and Computer Science*, IEEE, pp. 1–4, 2010

Zhi, Q. et al.: Toward dynamic image mosaic generation (2012)
Zhi, Q.; Cooperstock, J. R.: Toward Dynamic Image Mosaic Generation With Robustness to Parallax, in: IEEE Transactions on Image Processing, Vol. 21, pp. 366–378, 2012

Zhou, Y. et al.: MR Video Fusion: Interactive 3D Modeling and Stitching (2018)
Zhou, Yi; Cao, Mingjun; You, Jingdi; Meng, Ming; Wang, Yuehua,; Zhou, Zhong: "MR Video Fusion: Interactive 3D Modeling and Stitching on Wide-Baseline Videos", in: *Proceedings of the 24th ACM Symposium on Virtual Reality Software and Technology*, 2018

Zhou Wang et al.: Image quality assessment: from error visibility (2004)
Zhou Wang; Bovik, A. C.; Sheikh, H. R.,; Simoncelli, E. P.: Image quality assessment: from error visibility to structural similarity, in: IEEE Transactions on Image Processing, Vol. 13, pp. 600–612, 2004