Karlsruher Schriften
zur Anthropomatik

Band 54

Jürgen Beyerer, Tim Zander (Eds.)

Proceedings of the 2021 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory

KIT Scientific Publishing

Jürgen Beyerer, Tim Zander (Eds.)

Proceedings of the 2021 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory

Karlsruher Schriften zur Anthropomatik
Band 54
Herausgeber: Prof. Dr.-Ing. habil. Jürgen Beyerer

Eine Übersicht aller bisher in dieser Schriftenreihe
erschienenen Bände finden Sie am Ende des Buchs.

Proceedings of the 2021 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory

by
Jürgen Beyerer, Tim Zander (Eds.)

KIT Scientific Publishing

Impressum

Scientific Publishing

Karlsruher Institut für Technologie (KIT)
KIT Scientific Publishing
Straße am Forum 2
D-76131 Karlsruhe

KIT Scientific Publishing is a registered trademark
of Karlsruhe Institute of Technology.
Reprint using the book cover is not allowed.

www.ksp.kit.edu

Print on Demand 2022 – Gedruckt auf FSC-zertifiziertem Papier

ISSN 1863-6489
ISBN 978-3-7315-1171-7
DOI 10.5445/KSP/1000143483

Preface

In 2021, the annual joint workshop of the Fraunhofer Institute of Optronics, System Technologies and Image Exploitation (IOSB) and the Vision and Fusion Laboratory (IES) of the Institute for Anthropomatics, Karlsruhe Institute of Technology (KIT) was hosted for the second time due to the pandemic at the IOSB in Karlsruhe.

For a week from the 2nd to the 6th of August the PhD students of the both institutions delivered extended reports on the status of their research and participated in heated discussions on topics ranging from computer vision and optical metrology to usage control, control theory and neural networks. Most results and ideas presented at the workshop are collected in this book in the form of detailed technical reports. This volume provides a comprehensive and up-to-date overview of some of the research program of the IES Laboratory and the Fraunhofer IOSB.

The editors thank Arno Appenzeller, Paul Wagner and the other organizers for their efforts resulting in a pleasant and inspiring atmosphere throughout the week. We would also like to thank the doctoral students for writing and reviewing the technical reports as well as for responding to the comments and the suggestions of their colleagues.

Prof. Dr.-Ing. habil. Jürgen Beyerer
Dr. Tim Zander

Contents

Preface I
Jürgen Beyerer and Tim Zander

Maximum Entropy Consent Privacy Impact Quantification 1
Arno Appenzeller

Trustworthy Artificial Intelligence 21
Maximilian Becker

A Simple Pyramid Vision Transformer for Human Pose Estimation . 33
Mickael Cormier

Temporal Bird's Eye View for 3D Semantic Segmentation 53
Fabian Duerr

A Review on Deep Learning Approaches for Spectral Imaging 69
Benedikt Fischer

Dynamic Planning Pipeline for Indoor Inspection Flights 87
Raphael Hagmanns

Approaches for Causal Structure Identification 105
Josephine Rehak

Spectral Imaging for Stress Monitoring 121
Petra Schumacher

A Transformer-based Multi-task Model for Attribute-based Person Retrieval 139
Andreas Specker

Multi-Person Tracking with a Multi-Hypothesis Approach for Ambiguous Assignments 153
Daniel Stadler

Conceptualization of a Trust Dashboard for Distributed Usage Control Systems 169
Paul Georg Wagner

Cross-Domain Fine-Grained Classification 189
Stefan Wolf

Attention Mechanism in Computer Vision 207
Chengzhi Wu

Using Maximum Entropy to Extend a Consent Privacy Impact Quantification

Arno Appenzeller

Vision and Fusion Laboratory
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
arno.appenzeller@kit.edu

Abstract

Due to the progress of digitization in the medical sector digital consent becomes more and more common. While digital consent itself has a huge number of benefits for the researcher it can impose a lot of questions for the individual giving it. One of those questions is what impact the consent to sharing data with a research project has on the individual's privacy. The Consent Privacy Impact Quantification (CPIQ) provides a quantification to help the user making a consent decision based on the potential data sharing risk and his individual acceptance preferences for a research project. While this quantification provides a good first estimation it has some limitations especially in the method the re-identification risk is calculated for a member of a dataset. This paper presents a method using the Maximum Entropy principle. This principle provides a way to measure the maximum unbiased distribution using limited background knowledge, which is provided by epidemiological data. This distribution can then be used to see how much higher the re-identification risk based on a sensitive attribute is compared to the uniform distribution. In addition, the first promising results of the method will be shown based on an experimental setting.

1 Introduction

Through the ongoing digitization medical research has potential access to an enormous amount of data. The recently soft-launched German "Elektronische Patientenakte" (ePA) also offers functionality of a research platform. While the main purpose is to provide a safe and secure storage for health data that is created during medical treatment, such data could potentially be useful for medical research. Having access to the digital treatment records of millions of people could lead to huge benefits, such as Big Data analysis of medical data. Analyzing large scale data is one of the most promising techniques to improve future treatment and make huge progress in medical care. Besides the obvious benefits there remain open questions regarding the privacy of the processed data. The European General Data Protection Regulation (GDPR) considers medical data to be highly sensitive which is prohibited to process by default. But there are several exceptions where one is the explicit consent of the affected person. Currently, this is the usual way to use data of a patient in medical research. Through the digitization the paper-based consent is more and more replaced and first digital consent systems are coming close to productive use [2, 3, 4]. On the one hand digital consent makes giving consent easier but it does not necessarily make the actual decision easier. In fact, many parties or research projects to grant access can make the decision even harder. Additionally, every consent made for medical data should be an informed consent. While the definition of what an informed consent is remains a research topic on its own, there are systems needed that support patients when making consent decisions. Such a system is the Consent Privacy Impact Quantification (CPIQ) [1]. CPIQ provides a way to measure different properties of a research project and considers the potentially shared data to support the affected patient with their a consent decision. CPIQ considers many different properties but lacks an actual quantification of the shared data of an individual in regard to the database where it is shared to. Such a look can make a huge difference in terms of privacy because unique and striking data can make re-identification a lot easier. Unfortunately, such things are often only noticed after the data is added. This could be too late to protect the privacy and it is too late to avoid a risky sharing decision. In this paper we will use the Maximum Entropy principle to provide a conservative estimation of

the actual privacy risk of the data release. Therefore, we use epidemiological observations that are used as constraint for the Maximum Entropy methods. It is shown that using such methods can provide an accurate estimation how likely the re-identification of an individual is in a dataset. The remainder of this paper is structured as follows: Section 2 looks on related work of this topic. Section 3 introduces theoretical preliminaries that are needed to understand CPIQ and the method itself. Section 4 then describes our extension of CPIQ. Section 5 looks at our experiments. With this discussion the paper will be concluded and an outlook on future work will be provided.

2 Related Work

There is a multitude of papers that contribute to the topic of quantification of medical data in terms of privacy in various ways. Veeningen et al. describe a formal model for pseudonymization [14]. They use an exemplary digital health infrastructure where different data is shared across various sites. Each party is allowed to have different parts of data of an individual. The paper presents a so-called coalition graph which shows which party can combine which data to gain all data of a patient. This graph can be used to compare different data protection concepts. In contrast to this report Veeningen's approach focuses on pseudonymization architecture. While the idea for the formal model is very interesting this report considers the patient's view on its data and what impact on individual privacy data sharing has. The authors of "Quantifying the costs and benefits of privacy-preserving health data publishing" introduce a cost model for personal health records [9]. The approach tries to measure the cost of privacy and utility by comparing the cost of anonymization with the costs of a potential data breach. With the provided formulas a detailed comparison is possible but this approach is not suitable to measure individual data. Wan et al. look at a game theoretic approach to measure re-identification risks [15]. The authors try to weigh the factor between the monetary value of health data and the potential fine for a violation of privacy rules. They use different properties like generalization strategies for the data and their costs to create their model. An attacker is described that attempts re-identification when the benefit outweighs

the costs. The paper concludes that it is possible to find something like zero risk if there is no incentive to attempt an attack. A game theoretical approach is not compatible with the main idea of CPIQ which is to measure an individual risk and provide decision support when sharing personal health data. Additionally, our attacker tries to re-identify regardless of the costs to express that individual risk of re-identification. Another work by Wang et al. presents methods to measure the privacy level of a dataset [16]. Their method evaluates the privacy impact of data with quantitative and qualitative factors. The factors will be assessed using hierarchical decision making with the help of expert knowledge to rate data sensitivity. The result is then combined into a so-called privacy score. In contrast to our work the need for expert knowledge can be a high obstacle in terms of real-world execution. A paper that is closer to our approach is "Privacy-MaxEnt" by Du et al. [5]. The authors consider a scenario where quasi-identifiers are bucketized with sensitive attributes. An example is gender and age as quasi-identifiers and diseases as sensitive data. The main principle would be that the probability that a sensitive attribute belongs to the individual is distributed equally. It is shown that this is not the case for certain sensitive attributes (e.g., gender specific diseases). This background knowledge is then modeled by using Maximum Entropy to show the probability given the sensitives attributes. While the paper discusses a sophisticated approach it lacks a real use case. It also makes no decision on what background knowledge should be used to model the constraints. In our work we use the core idea of the paper and extend our CPIQ technology with concrete examples. None of the here presented approaches describe a complete quantification that can support the decision of an individual to share its personal health data.

3 Preliminaries

In this section the preliminaries needed for MaxEnt CPIQ are described. We first introduce some common privacy preserving techniques which are used in CPIQ and explain the motivation to mitigate re-identification attacks. MaxEnt CPIQ uses Maximum Entropy to provide a more accurate privacy impact quantification. The concept of Maximum Entropy will be also explained in this section. Finally,

the formal consent model needed for CPIQ is introduced and CPIQ itself is explained.

3.1 Privacy Preserving Technologies

The motivation behind the consent privacy impact quantification of CPIQ is to provide the affected person with an estimation of what risk comes with sharing its medical data. Even if the data is anonymized or pseudonymized before it is given to a third party there is still the risk of re-identification with background knowledge. Obvious examples are a large data set where only one person has a very rare disease. Such cases can be easily identified. However, there are several more sophisticated re-identification approaches that were shown in several studies and go beyond purely academic examples [12, 11]. Considering personal health data as highly sensitive data it should be clear that measures are needed to mitigate this risk. Therefore, different privacy preserving technologies exist. Besides technologies like homomorphic encryption, which is used in more and more cases, or statistical guarantees like differential privacy (DP) more traditional approaches rely on suppressing or generalizing quasi-identifiers. One of them is k-anonymity which requires that in a dataset there needs to be at least $k-1$ other individuals with the same quasi-identifiers [13]. This helps to reduce re-identification based on background knowledge about quasi-identifiers which could be de-facto public knowledge. To reach this goal suppression, where quasi-identifiers are removed, or generalization, where quasi-identifiers are grouped into more general categories, is used to form equivalence classes. One weakness of k-anonymity is that it does not consider the sensitive attribute itself. This is where l-diversity comes into place [10]. l-diversity requires that at least $l-1$ distinct sensitive attributes exist in each equivalence class. This mitigates the risk for cases where re-identification would be trivial. For example, k-anonymity would allow equivalence classes where everyone has the same sensitive attribute. This would be a privacy leak itself.

3.2 Maximum Entropy

The principle of maximum entropy follows the idea to define a maximum unbiased distribution given some constraints, which would be the distribution with the largest entropy. This information theory concept itself was introduced in 1957 by Jaynes [7, 8]. What is called constraints above can also be described as testable information. This information gives a mathematical statement of the probability distribution. A testable information can be that the sum of two event probabilities p_1 and p_2 is smaller than 0.5. Depending on the definition of Maximum Entropy there always is the universal constraint that the sum of all event probabilities is 1. Given those constraints equations can be formed under that the distribution fulfills the constraints and maximizes entropy. To solve the equations the so-called Lagrange multipliers can be used. Those mathematical details can be found in the original publication and are not looked at in this paper.

3.3 Formal Consent Model

The foundation for CPIQ is the formal consent model which was introduced in the original publication. The formal model defines the properties required to describe consent for secondary usage (e.g., research). The model is based on the technical consent model of the German ePA and is combined with properties out of the research consent template of the German "Medizin Informatik Initiative" which is widely accepted by data regulation authorities. Table 3.1 gives an overview of the properties. It consists of the subject S who can be a patient or a legal guardian. The researcher is referenced as the authorized party AP. Every declaration of consent is required to have a timespan $TS = (TS_{Start}, TS_{End})$ during it is valid. The consent then is defined through policies which also contain which documents or categories R are shared. A full policy consists out of P = [(AP, TS, R, A)] where A is the action allowed on the resources. For secondary usage this is limited to a read action. The next properties are focused on the concrete research project. The purpose PU of a project is considered as well as potential personal or social benefits PBE and SBE which are listed in BE. Furthermore, the degree of anonymization DA_D and

Identifier	Explanation
S = Patient | Legal Guardian	Subject
AP = [Researcher]	Authorized Party
TS_{Start} = Date	Starting Date
TS_{End} = Date	End Date
TS = (TS_{Start}, TS_{End})	Timespan
R = [Document | Category]	Resource
A = Read (r)	Action
P = [(AP, TS, R, A)]	Policies
PU = [Purpose] | Broad Consent	Research purpose
PBE = [Personal Benefit]*	Personal Benefit
SBE = [Social Benefit]*	Social Benefit
BE = [PBE | SBE]*	Benefit
DA_D = (k-Anonymity, l-Diversity)	Degree of anonymization for D
PS = Low | Medium | High	Processing security
D = (PS, DA_D)	Data processing
DA_{PUB} = (k-Anonymity, l-Diversity)	Degree of anonymization for P
I = ($false$ | $true$)	Information
PUB = (($false$ | $true$), DA_{PUB})	Publication
T = (I, PUB)	Transparency
RI = (PU, BE, D, T)	Research information

Table 3.1: Identifiers for the formal consent model

the processing security PS are part of the data processing $D = (PS, DA_D)$ of a project. Another factor is transparency T which consists out of information value I and the publication value PUB which states if there is a publication and if yes what kind of anonymization DA_{PUB} is used. This is then listed in the research information $RI = (PU, BE, D, T)$.

3.4 Consent Privacy Impact Quantification (CPIQ)

Based on the formal consent model mentioned in Section 3.3 CPIQ provides a consent privacy impact quantification that consists out of two main parts: acceptance and risk of a consent decision. The idea is that CPIQ calculates a score (in most times from 0-100) which indicates the higher it is the more the acceptance factors (AF) outweigh the potential risks. For the acceptance factors we refer to the original publication. They consist of the user-weighted formal model properties of purpose, personal and social benefits, information, publication, and trust.

The risk will be explained in more detail because this is where our model extension comes into place. At first, we assume an attacker that has access to all publicly available data of a patient. The attacker's goal is to gain knowledge about a potential sensitive attribute of an individual. This goal can be reached through a classical re-identification attempt. To quantify this risk two main attack points are identified. The first is the attack on the stored private data of the research project. For this the risk probability of a data breach DLP also needs to be considered. The second way is to try re-identification on the data that are part of the published data of a project. This depends on the publication factor probalitity PF. In both cases the re-identification itself will be measured through the sensitive attribute exposure probability $SAEP$. The assumption for CPIQ is that every project uses a certain degree of l-diversity as privacy preserving technology. So $SAEP = Min(1, \frac{|R|}{l})$ with $|R|$ as number of resources an individual has in the dataset. We also assume that a patient can have more than one sensitive attribute in the dataset, so this "row" in a dataset is mapped to the same set of quasi-identifiers and will weaken the l-diversity property. This leads to $\frac{|R|}{l}$ as probability. If there are more properties than ensured through l-diversity re-identification is

obvious and therefore the probability is one. This combined with the two attack vectors results in the re-identification probability due to data leakage $RPD = P(\text{Damage}_{\text{Data leakage}}) = DLP * SAEP_{DP}$ and re-identification probability due to publication $RPP = P(\text{Damage}_{\text{Publication}}) = PF * SAEP_{PUB}$. CPIQ also considers the location of processing as a risk factor and requires that all processing locations have a regulation with similar standards as GDPR which is defined by the GNF factor. The Total Re-Identification Risk Probability $TRRP = P(\text{Damage}_{\text{Total}}) = 1 - ((1 - RPD) * (1 - RPP) * (1 - GNF))$ is the result of this. Combined with the acceptance factors the CPIQ score will be calculated with the following equation:

$$CPIQ = AF * (\frac{L}{2} * (1 - \frac{1}{s})) + (1 - TRRP) * (\frac{L}{2} * (1 + \frac{1}{s}))$$

with $s \geq 1$ as weighting factor between risk and acceptance and L as maximum value.

4 Using Maximum Entropy with CPIQ

After we introduced the preliminaries, this section will point out why the extension with Maximum Entropy should be done and how it can be implemented. In addition, we show some experiments with the new approach.

4.1 Model extension

Section 3.4 described the factors used to calculate the risk for CPIQ. One assumption made is that l-diversity is used as privacy preserving technology. This is used for $SAEP$ which is one of the main factors. While this assumption can be made it is not clear that this always can be implemented in practice. While anonymization and pseudonymization technologies are a common practice in medical research, it does not seem too realistic that every research project uses a suitable degree of l-diversity or that l-diversity can be applied in a good way. This could lead to the case that CPIQ does not give a very accurate consent evaluation. The goal for the extension was to consider more the uniqueness of

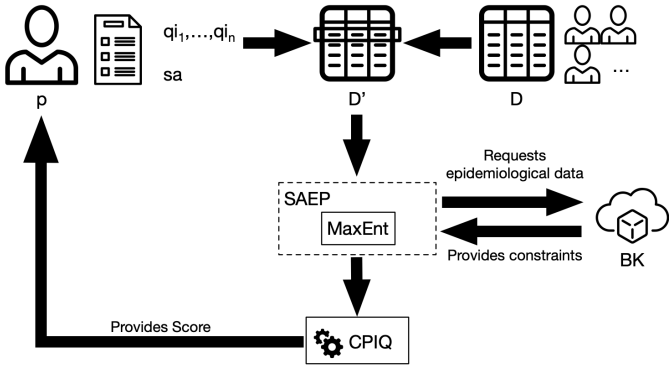


Figure 4.1: MaxEnt CPIQ Workflow

a sensitive attribute of an individual. While l-diversity provides a method for this on a database level it does not consider the background knowledge of a potential attacker. It also requires the assumption that every sensitive attribute is uniformly distributed. Especially with medical data this is not the case. There are certain diseases that are more common than others. In addition, different factors like age or gender can heavily affect the frequency of a disease. An obvious but good example for this is breast cancer, which occurs in female and male persons. However, breast cancer is very rare for males so that a database with different cancer types from individuals with both genders could lead to an easier linking of the sensitive attributes to the individuals. We found that Maximum Entropy suits best to include such information. This also deals with the facts that no background knowledge can be complete. For this the Maximum Entropy principle provides the best non-biased estimation given the currently available information.

To replace $SAEP$ with l-diversity an individual $p = (\{qi_1, ..., qi_n\}, sa)$ is introduced. The individual wants to give its sensitive attribute sa with its quasi-identifiers to a dataset D, which already includes other patients with sensitive data. We also introduce a background knowledge source BK which uses publicly available medical information like disease incidence per gender, age, region, and more epidemiological data to provide background knowledge

constraints bkC_i. Figure 4.1 shows the workflow for Maximum Entropy CPIQ. The individual p provides its data to be combined before sharing with the dataset D of the research project. It remains to be noted that it is an open question where this combination and processing happens. One option would be to do it locally at the patient's device or at a trusted third party. This combined data set D' will be then used to calculate the $SAEP$. Therefore, epidemiological data for the given sensitive attributes and quasi-identifiers will be requested from BK. In our case this data is the incidence for the given diseases per gender and per region. This incidence is then used as constraints for a Maximum Entropy distribution. To calculate the risk the difference between the uniformly distributed re-identification risk of all individuals is compared with the constrained Maximum Entropy distribution re-identification risk. This factor is then divided by a custom threshold. This risk threshold defines the factor of how much higher the risk can be tolerated compared to uniformed distribution. The minimum of the received value or 1 will be returned as $SAEP$ value and the rest of the CPIQ process can continue as described.

Definition 4.1.1 (MaxEnt CPIQ)**.** Let UD be the uniform distribution of D. uR is the share of any individual in the distribution ($uR = 1/n$) where n is the number of individuals in the dataset. CD is the constrained distribution of D. This is calculated by using the given constraints for D with the Maximum Entropy principle. cR_i is the constrained distribution of a given individual p_i. The personal risk factor is then $pRF_i = cR_i/uR$. Let $\perp$ be the risk threshold. The weighted risk ratio is then $rr_i = min(1, pRF_i/\perp)$. rr_i is a value between 0 and 100% so that 1 (100%) is the maximum.

4.2 Experiments

To show the feasibility of the extension some experiments were done. Some exemplary scenarios with small sample datasets were created to show the approach in a comprehensible way. For this the technique was implemented in Python[1]. Maximum Entropy was implemented by using the Python Package

[1] https://www.python.org

ICD10 C*	C00-C14	C15	C16	C18-C21	C22	C50	...
Male Overall	17.2	9.0	14.8	51.5	9.4	0.7	...
Female Overall	6.9	2.2	7.5	35.1	3.6	109.2	...

Table 4.1: Exemplary excerpt of incidence data for different cancer types as ICD-10 Codes per gender

maxentropy[2]. The scenario is a database with of several individuals that have different types of cancer. For the background knowledge data on cancer we used the population wide cancer incidence provided by the German Center for Disease Control the Robert Koch Institut [6]. Table 4.1 shows an excerpt of the aggregated data. The actual data set also includes age specific and region-specific incidences. For this scenario we only differentiated by a higher risk age for breast cancer (ICD-10 code C50; higher risk with age older than 60) and the lower risk age. Since in our dataset every person has a disease and the incidence is a population wide metric, we also calculated a total share depending on the incidence and the complete data set. The labels in the dataset have the format: (qi_1, sa, qi_2) where qi_1 is the gender, qi_2 is the age and sa is the ICD-10 code for the type of cancer. As risk threshold $\perp = 3$ is used.

4.2.1 Scenario 1: Adding a lower risk person

Figure 4.2 shows the constrained distribution of the scenario dataset D before the additional subject is inserted. The male person with breast cancer (C50) has a very low risk to be relinked to its sensitive attribute which is obvious because it is rare for males. In addition, the individual with prostate cancer (C61) has a very high risk since this is one of the most common cancer types for males. Furthermore, this disease does not exist for females because of biological reasons. Next a new subject wants to share its data. The data is from a female with breast cancer in the lower risk age range $p_1 = (”Female”, ”50”, 40)$. Figure 4.3(a)

[2] `https://github.com/PythonCharmers/maxentropy`

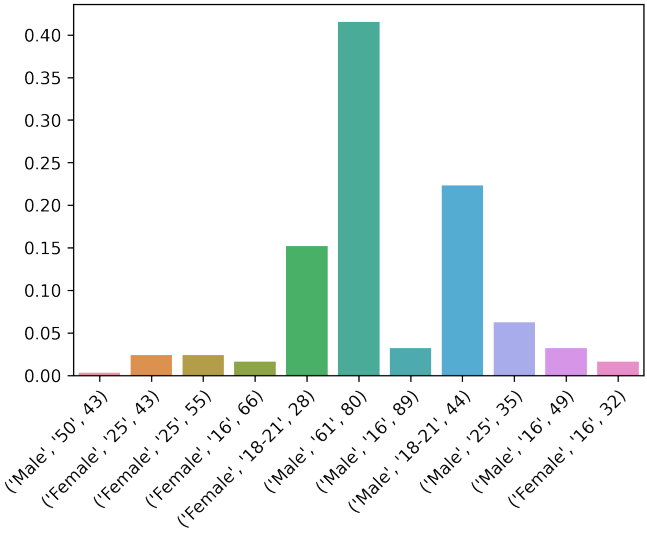
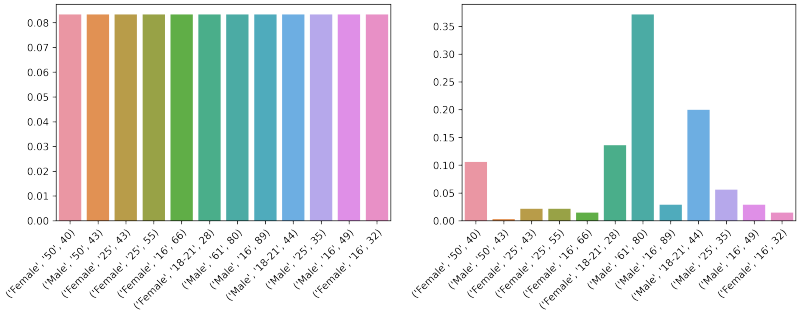


Figure 4.2: Scenario Dataset D before insertion with constrained distribution



(a) Uniform distribution

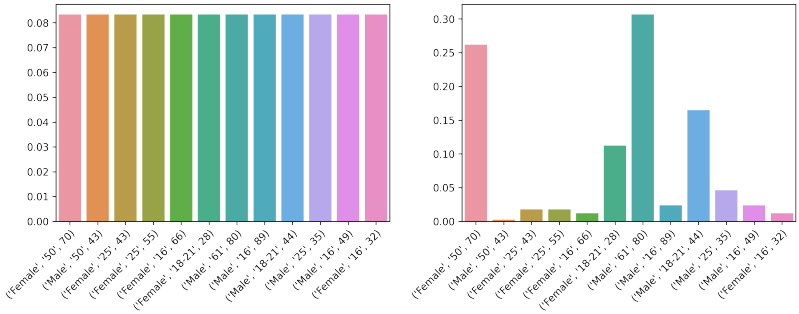
(b) Constrained distributed

Figure 4.3: Scenario 1 Dataset D' after insertion of p_1

shows the dataset with the inserted data and an assumed uniform distribution. This is needed to calculate the difference between the constrained distribution that can be seen in Figure 4.3(b). The constrained distribution shows that the re-identification is higher than with the uniform distribution but only slightly. The weighted risk ratio for $SAEP$ is 0.32, which can be considered as lower risk.

4.2.2 Scenario 2: Adding a higher risk person

The starting situation is the same as in Scenario 1. This time a higher risk for breast cancer person $p_2 = (”Female”, ”50”, 70)$ is added. Figure 4.4(a)



(a) Uniform distribution

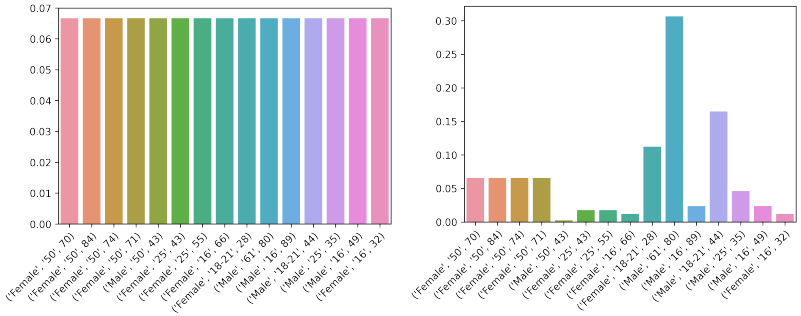
(b) Constrained distributed

Figure 4.4: Scenario 2 Dataset D' after insertion of p_2

shows the uniform distribution and Figure 4.4(b) the constrained distribution. The risk is much higher than for the low-risk person. In fact, this has now the second highest risk which can also be seen in the $SAEP$ risk ratio which is the maximum with 1.

4.2.3 Scenario 3: Adding three higher risk persons

In contrast to Scenario 2 it needs to be looked at what happens when risk is more evenly distributed. Therefore, three higher risk persons are added. Figure



(a) Uniform distribution

(b) Constrained distributed

Figure 4.5: Scenario 3 Dataset D' after insertion of three higher risk persons

4.5(a) and 4.5(b) show that the data sharing for the individual higher risk person has now a smaller risk than before. The individual risk ratio for one of the three persons is $0,33$ which is a bit higher than in Scenario 1 but much smaller than in the second experiment.

5 Discussion

The experiments in Section 4.2 showed a proof of our concept. However, the experiment was done with a small sample size and is no complete proof for the principle. Nevertheless, Maximum Entropy provides a good way to include background knowledge. Publicly available data like the cancer registry data can be included easily as constraints for a re-identification metric. While more traditional metrics like k-anonymity or l-diversity provide a concrete way for the data owner to improve the privacy impact of a dataset those values remain

vague for the affected person. In addition, not every dataset can implement any value for k or l. Furthermore, the generalization or suppression to receive several specific sized equivalence classes is no trivial task. From this point of view the Maximum Entropy model has fewer requirements depending on the data structure. Instead, it requires a specific form of data input for the model. Any database should be able to be mapped to the format of quasi-identifiers and the sensitive attribute which should be measured for uniqueness in the dataset. Public databases to form constraints out of epidemiological data should be also widely available. On the other hand, there are open questions where to process the evaluation. Expecting the individual to let the data process by a third party could easily require the same level of trust as it would to give the consent and share the data with the researcher. Another idea would be to provide the current dataset D to the potential participant to do the CPIQ evaluation. This could be unacceptable for research institutes for privacy reasons or even for intellectual property reasons. A trusted third party by the potential participant and the researchers would solve this but it would require high standards to gain this trust. While there were no user studies, we think that our risk calculation is more natural by using a metric that considers how much higher the risk depending on the quasi-identifier and epidemiological background knowledge for a sensitive attribute is compared to if every sensitive attribute is distributed equally in the population. As our experiments show there can be a hen egg problem with smaller datasets or rare diseases. While adding one individual that has a high risk for breast cancer would lead to bad CPIQ recommendation from the risk side it would be better if there were three persons with the same sensitive attribute. This imposes the questions where the additional persons should come from and if it is ok to assume that there are some privacy risk friendly persons that share their data regardless of the CPIQ score. The same applies to small datasets. Another thing that should be noted is that our experiments are limited. There is no complete comparison against the l-diversity version of $SAEP$ or an evaluation with real world data.

6 Conclusion and Outlook

This technical report presents an extension to the risk model of the consent privacy impact quantification CPIQ. The original form of CPIQ uses l-diversity to measure the individual privacy risk for a patient that wants to share his data. This imposes many issues and may be an impracticable requirement. Therefore a method that does not impose any requirements on the data structure or certain anonymization methods was needed. The Maximum Entropy principle is a promising method for this. It can be used to measure a maximum unbiased distribution based on limited background knowledge. As source for background knowledge epidemiological data which is publicly available for the potential disease as sensitive attribute is suggested. With this the Maximum Entropy principle can be used to measure the difference between a uniform distribution and the constrained distribution. This difference can then be used as weighted risk ratio which replaces l-diversity in the CPIQ method. An experimental evaluation is presented, and the results are discussed. While this paper does not provide a complete analysis of this extension the first results look very promising, and the Maximum Entropy extension seems to be a feasible method with less requirements than the original method.

As described before this paper does not provide a full analysis of our suggested extensions. For future work a complete evaluation against l-diversity is needed. It is important to measure the difference between l-diverse tables and their results in $SAEP$ and when the Maximum Entropy principle is used on this data. A limitation would be that not every assumption that was made for the original form of CPIQ can also be made for the extension. This also needs to be analyzed in depth. Furthermore, a real-world dataset evaluation would be very interesting. Our experiments only used a very limited and small sample data set. It would be interesting to obtain a real-world data set from for example a hospital or a cancer registry and measure the constrained distribution in this dataset. This could also be used to analyze the acceptance of such a method. Finally, the question for an optimal distribution and size of a dataset should be looked at. Our experiments showed that the size of a dataset and the distribution of it has a large influence on the risk estimation. While this is a question itself

the here introduced method can be used to recommend optimal datasets that have a high acceptance for the potential data donors.

References

[1] Arno Appenzeller et al. “CPIQ - A Privacy Impact Quantification for Digital Medical Consent”. In: *The 14th PErvasive Technologies Related to Assistive Environments Conference*. PETRA 2021. New York NY, USA: Association for Computing Machinery, 2021, pp. 534–543.

[2] Arno Appenzeller et al. “Enabling Data Sovereignty for Patients through Digital Consent Enforcement”. In: *Proceedings of the 13th ACM International Conference on PErvasive Technologies Related to Assistive Environments*. PETRA ’20. Corfu, Greece: Association for Computing Machinery, 2020. ISBN: 9781450377737. DOI: 10.1145/3389189.3393745. URL: https://doi.org/10.1145/3389189.3393745.

[3] M. Bialke et al. “MOSAIC - A Modular Approach to Data Management in Epidemiological Studies”. In: *Methods of Information in Medicine* 54.04 (2015), pp. 364–371.

[4] M Bialke et al. “A workflow-driven approach to integrate generic software modules in a Trusted Third Party.” In: *J Transl Med* 13 (2015), p. 176.

[5] Wenliang Du, Zhouxuan Teng, and Zutao Zhu. “Privacy-MaxEnt: Integrating Background Knowledge in Privacy Quantification”. In: SIGMOD ’08. Vancouver, Canada, 2008.

[6] Robert Koch-Institut (Hrsg) und die Gesellschaft der epidemiologischen Krebsregister in Deutschland e.V. *Krebs in Deutschland fr 2015/2016. 12. Ausgabe*. In German. 2019.

[7] E. T. Jaynes. “Information Theory and Statistical Mechanics”. In: *Physical Review* 106.4 (May 1957), pp. 620–630. DOI: 10.1103/PhysRev.106.620.

[8] E. T. Jaynes. “Information Theory and Statistical Mechanics. II”. In: *Physical Review* 108.2 (Oct. 1957), pp. 171–190. DOI: 10.1103/PhysRev.108.171.

[9] RH Khokhar et al. “Quantifying the costs and benefits of privacy-preserving health data publishing.” In: *J Biomed Inform* 50 (2014), pp. 107–121.

[10] A. Machanavajjhala et al. “L-diversity: privacy beyond k-anonymity”. In: *22nd International Conference on Data Engineering (ICDE’06)*. 2006, pp. 24–24. DOI: 10.1109/ICDE.2006.1.

[11] Yves-Alexandre de Montjoye et al. “Unique in the Crowd: The privacy bounds of human mobility”. In: *Scientific Reports* 3 (2013).

[12] Arvind Narayanan and Vitaly Shmatikov. “How To Break Anonymity of the Netflix Prize Dataset”. In: *CoRR* abs/cs/0610105 (2006). arXiv: cs/0610105. URL: http://arxiv.org/abs/cs/0610105.

[13] Latanya Sweeney. “k-Anonymity: A Model for Protecting Privacy”. In: *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.* 10.5 (Oct. 2002), pp. 557–570. ISSN: 0218-4885. DOI: 10.1142/S0218488502001648. URL: https://doi.org/10.1142/S0218488502001648.

[14] Meilof Veeningen, Benne de Weger, and Nicola Zannone. “Formal Modelling of (De)Pseudonymisation: A Case Study in Health Care Privacy”. In: *Security and Trust Management: Lecture Notes in Computer Science*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 145–160.

[15] Z Wan et al. “A game theoretic framework for analyzing re-identification risk.” In: *PLoS One* 10.3 (2015), e0120592.

[16] Dan Wang, Bing Guo, and Yan Shen. “Method for measuring the privacy level of pre-published dataset”. In: *IET Inf. Secur.* 12.5 (2018), pp. 425–430.

Trustworthy Artificial Intelligence

Maximilian Becker

Vision and Fusion Laboratory
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
maximilian.becker@kit.edu

Abstract

Trust or trustworthiness are hard to define. There are many aspects that can increase or decrease the trust in an Artificial Intelligence systems. This is why entities such as the High-level expert group on AI (HLEG) and the European commission's artificial intelligence act are putting forward guidelines and regulations demand trustworthiness and help to better define it. One aspect that can increase the trust in a system is to make the system more transparent. For AI systems this can be achieved through Explainable AI or XAI which has the goal to explain learning systems. This article will list some requirements from the HLEG and the European artificial intelligence act and will go further into transparency and how it can be achieved through explanations. At the end we will cover personalized explanations, how they could be achieved and how they could benefit users.

1 Introduction

Trust and trustworthiness are complex and not easy to define concepts. An organization that focuses on trustworthy artificial intelligence is the High-

level expert group (or HLEG) on artificial intelligence[1]. The HLEG was appointed by the European Union to advise on their artificial intelligence strategy. They released an ethics guideline for trustworthy AI in which they define seven requirements for trustworthiness in AI systems[2]: 1) human agency and oversight, 2) technical robustness and safety, 3) privacy and data governance, 4) transparency, 5) diversity, non-discrimination and fairness, 6) environmental and societal well-being and 7) accountability. The HLEG has the aspiration to shape the EU's future approach to AI. With their ethics guideline they took their first step to make AI more trustworthy by listing their requirements. In this article we are going to focus on transparency.

The European Union also put forward a draft for a regulation called the artificial intelligence act which should enable the development and deployment of trustworthy AI. The new regulation is called the artificial intelligence act and will be covered in section 2. One way to increase the transparency of AI systems is through explainable AI or XAI. The goal of this field is to generate explanations for learning systems. Section 3 will give an overview over the field and how these explanations can look. Afterwards section 4 describes five different concrete approaches to explainability and give examples for each approach. Finally section 5 will look into personalizing these explanations, which means that the explanations are adapted to a users wants and needs, to make them more relevant for individual users.

2 Artificial Intelligence Act

The Artificial Intelligence Act or AI-Act for short is a draft for a regulation from the European commission from 2021. The full name is: Proposal for a Regulation laying down harmonised rules on Artificial Intelligence (Artificial Intelligence Act) and amending certain Union legislative acts[4]. It is a legal framework similar to the GDPR [5] but written specifically for AI systems and

[1] High-level expert group on artificial intelligence, https://digital-strategy.ec.europa.eu/en/policies/expert-group-ai

[2] Ethics guidelines for trustworthy AI, https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai

should lay the groundwork for trustworthy AI. The technologies falling under the new regulation are defined as [4]:

- Machine learning approaches, including supervised, unsupervised and reinforcement learning, using a wide variety of methods including deep learning"
- "Logic- and knowledge-based approaches, including knowledge representation, inductive (logic) programming, knowledge bases, inference and deductive engines, (symbolic) reasoning and expert systems"
- "Statistical approaches, Bayesian estimation, search and optimization methods.

The regulation puts forward transparency rules for AI systems used in these applications[4]:

- Interaction with humans
- Emotion detection
- Biometric identification
- Generation and manipulation of content such as Deep fakes

The regulation defines four risk levels: unacceptable risk, high risk, and low or minimal risk. Applications that fall under the first level are mostly prohibited. They are for example [4] systems that exploit vulnerabilities such as disabilities and are likely to cause physical or mental harm, real-time remote biometric identification in public places for law enforcement, or systems that use subliminal techniques beyond a persons consciousness and may cause physical or psychological harm. The focus of the regulation is on high risk applications for which it poses strict requirements. Under this category are [4] systems used as safety components in other systems as well as ones used in some critical areas such as biometric identification, management of critical infrastructure, education and vocational training [4]. The last two levels, low and minimal risk are not further defined and the implementation of the regulation is on a voluntary basis.

The regulation lists requirements for high risk systems like a risk management system[4], testing procedures, technical documentation, transparency rules and others. The transparency rules state that the systems operation is sufficiently transparent to enable users to interpret the systems output and use it appropriately[4]. These transparency criteria can be achieved through XAI. Explainability research focuses on the one hand on making the output of systems more interpretable and how to present these explanations to the user. On the other hand it focuses on explaining the inner workings of models in order to better understand the models, their scope and their boundaries which enables users to use the systems appropriately. So XAI technologies are perfectly suited to fulfill these transparency requirements.

3 Overview over XAI

Many modern learning approaches like deep neural networks are very powerful but they are black boxes to developers and users. This means that the models are very good at making predictions but it is not clear how they make their decisions. Explainable Artificial Intelligence or XAI has the goal to explain the decisions of learning systems. To do this there are generally 2 approaches: explain existing methods like deep neural networks or use inherently explainable approaches. Both approaches have their advantages and disadvantages. In the first case any model can be used and the explanation can be generated post-hoc. This means that no compromise on the used model necessary. However, these explanations often explain only part of the model or an approximation of it. So it can happen that the explanation is only some kind of artifact that is not really present in the model or data. The second approach uses models that are inherently explainable. This means that the model itself can be understood and not only an approximation or part of it. But there is often a trade-off between predictive ability of a model and its explainability. This means that an explainable model is in general less powerful than a black box model. To illustrate this, we can compare a deep neural network to a small decision tree. The neural network will make much better predictions but it is not clear how or why it makes these predictions. The decision tree on the other hand is completely intelligible because one can just

check every node on the path that lead to a certain prediction but this comes at the cost of the models predictive power. A small decision tree will not be able to represent a complex classification problem.

Different explanation methods can be distinguished further[3]. An explanation can be global or local. Global explanations explain a whole model while local explanations only explain single predictions of the model. They can also be model agnostic or model specific. Model agnostic explanation methods can be used to explain any model while model specific explanations can only explain one or several specific models.

Another distinction is the dependence on training data. Explanations can be data dependent, which means that they need to be trained with data that is usually the training data of the model they should explain. They can also be data independent, which means that they only need access to the model itself or its predictions.

4 Approaches to XAI

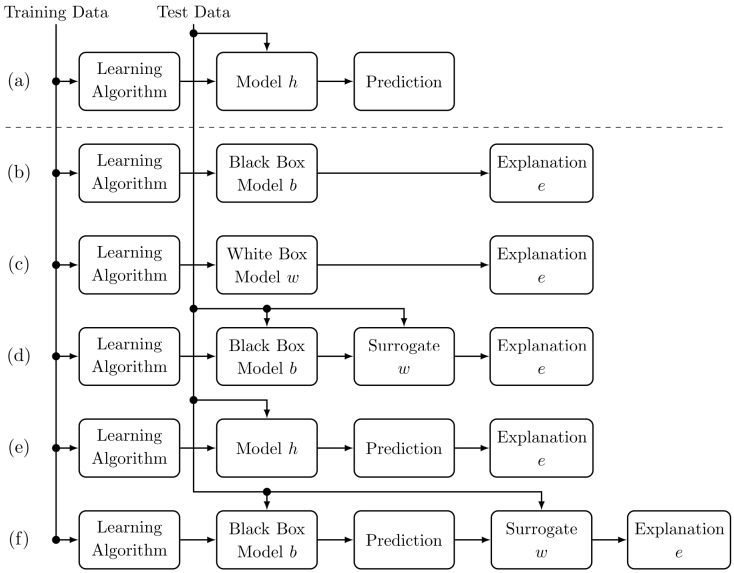


Figure 4.1: Default supervised learning and five different approaches to explainability[3] a) supervised machine learning, b) post-hoc explainability, c) white box model, d) global surrogate model, e) direct local explanation, f) local surrogate model

According to Burkart et al.[3] there are five different approaches to explainability for supervised machine learning (figure 4.1 (a)). The first approach is post-hoc explainability (figure 4.1 (b)). Here a black box model is trained on the training data and an explanation method is applied to the model afterwards. This is a global approach because the model as a whole gets explained. Post-hoc explanations have the advantage that any model can be used to make predictions which ensures a high prediction accuracy. An example for post-hoc explanations are partial dependence plots [7]. They visualize the dependence of the prediction on different features.

The second approach are white box models (figure 4.1 (c)). This approach uses a white box model which is a model that is inherently explainable. Because the whole model is understandable this is also a global approach. However these models can suffer from the afore mentioned trade-off between predictive power and explainability. Examples are decision trees and the explainable boosting machine[10].

The next approach are global surrogates (figure 4.1 (d)). A surrogate is a replacement for a black box model that is more explainable. At first a black box model is trained and with the black box model and the training data a second surrogate model is trained. The black box model is used for predictions and the surrogate model is used to generate explanations. Because the whole surrogate is explainable this is also a global approach. An advantage is that the prediction accuracy is conserved because a black box model is used to make the predictions. But a problem of this approach is that the surrogate model is only an approximation of the black box model so there will be a difference in the predictions of the two models, if they were identical the surrogate could be used as a white box model. This difference means that explanations generated with the surrogate only approximate the decisions made by the black box model. As an example decision trees can be used as a surrogate to approximate another model and explain it.

The fourth approach are direct local explanations (figure 4.1 (e)). Here a model is trained and used to make predictions. These predictions are then explained. Because only individual predictions and not the whole model are explained this is a local approach. An example are counterfactual explanations[14] which explain the decision for one instance by providing a second instance that leads to a different, desired prediction. So the counterfactual is another instance from the feature space that lies past a decision boundary and is ideally close to the original instance. The two instances or just the difference between them can then be used as the explanation because they represent the change in the feature space that leads to a different prediction. For example if a credit application got denied a counterfactual explanation could be that the application would have been accepted if the credit amount was 1000 less.

The last approach are local surrogates (figure 4.1 (f)). They are similar to global surrogates but as the name suggests they are local explanations. A black box

model is trained and used to make predictions. A local surrogate model is then trained on samples from the area around the prediction and used to generate explanations. This surrogate does not represent the whole black box model but just the decisions in the vicinity of the instance of interest. An example are local interpretable model-agnostic explanations or LIME[12]. To generate the explanation points around the instance to be explained are sampled and then labeled using the black box model. A linear model is then trained with these samples under consideration of their distance from the original instance. This local model then represents the black box model's decisions in the vicinity of the instance and can be used to explain which features contributed more or less to the decision for the instance.

5 Making Explanations Personalized

Some research exists on what kind of explanations are suitable for which target groups. Different target groups like developers, domain experts and end users have different demands and levels of understanding and this must be considered when choosing or developing explanations[2].

A great way to improve XAI methods is to make explanations more personal and adapt them not only to groups of people but to the individual users. This would probably benefit end users the most as they are more interested in the decisions and their consequences to their lives than in understanding the model that made them. Personalized explanations would make the results more relevant to the user because they are adapted to their individual needs, requirements or preferences. This makes them easier to apply and may lead to more trust in the system.

We are going to look at ways to personalize counterfactual explanations (see section 4) from here on. A first step is to make them actionable. This means that only features that are easily changeable by the user are considered in the explanation, so for example the gender a person will not be regarded in the counterfactual instance. This is already done[1, 11] but does not really consider a persons preferences only world knowledge. A next step would be to not only exclude features but weigh the remaining ones. This would enable a system to

incorporate user preferences in much more detail. So for example a user may be able to change her job or salary rather easily but changing her residence is really hard. Such preferences could be incorporated into counterfactual explanations using a weighted distance metric in the search process[9]. Another possibility to make counterfactual explanations more realistic is to consider interdependence between features. For example getting a better education takes time so this will result in the person getting older. It is often said that counterfactual explanations should be sparse[6, 8, 9, 13]. This means that as few features as possible are changed. Different people can however also have different preferences on whether it is better to change a single feature by a lot or multiple features a little. This can also be considered when personalizing counterfactuals.

All these options give the user very detailed options to personalize her explanations. The drawback of this is that each user has to take action and setup their personal preferences. This may deter users from using such a system. A solution could be to make the process interactive. At first a generic explanation is generated and afterwards the user can tell the system that a certain feature should be changed less or not at all. This process could be repeated until a satisfying explanation is found. The data gained in this process could also be used to learn a users preferences and use them to improve future explanations.

6 Summary

In this paper we looked at different aspects of making AI systems more trustworthy. At first the HLEG on AI and the European Artificial Intelligence Act were presented. Afterwards we focused on explainability of AI systems in general and on different approaches to it. At the end a research proposal for personalized explanations was presented.

References

[1] André Artelt and Barbara Hammer. “Convex optimization for actionable\ plausible counterfactual explanations”. In: *arXiv e-prints* (2021), arXiv–2105.

[2] Alejandro Barredo Arrieta et al. “Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI”. In: *Information Fusion* 58 (Dec. 2019). DOI: 10.1016/j.inffus.2019.12.012.

[3] Nadia Burkart and Marco F Huber. “A survey on the explainability of supervised machine learning”. In: *Journal of Artificial Intelligence Research* 70 (2021), pp. 245–317.

[4] European Commission. *Proposal for a REGULATION OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL LAYING DOWN HARMONISED RULES ON ARTIFICIAL INTELLIGENCE (ARTIFICIAL INTELLIGENCE ACT) AND AMENDING CERTAIN UNION LEGISLATIVE ACTS*. URL: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A52021PC0206.

[5] European Commission. *Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation)*. URL: https://eur-lex.europa.eu/eli/reg/2016/679/oj.

[6] Susanne Dandl et al. “Multi-Objective Counterfactual Explanations”. In: *Lecture Notes in Computer Science* (2020), pp. 448–469. ISSN: 1611-3349. DOI: 10.1007/978-3-030-58112-1_31. URL: http://dx.doi.org/10.1007/978-3-030-58112-1_31.

[7] Jerome H Friedman. “Greedy function approximation: a gradient boosting machine”. In: *Annals of statistics* (2001), pp. 1189–1232.

[8] Thibault Laugel et al. “Inverse Classification for Comparison-based Interpretability in Machine Learning”. In: *stat* 1050 (2017), p. 22.

[9] Ramaravind K Mothilal, Amit Sharma, and Chenhao Tan. “Explaining machine learning classifiers through diverse counterfactual explanations”. In: *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. 2020, pp. 607–617.

[10] Harsha Nori et al. “InterpretML: A Unified Framework for Machine Learning Interpretability”. In: *arXiv preprint arXiv:1909.09223* (2019).

[11] Rafael Poyiadzi et al. “FACE”. In: *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society* (Feb. 2020). DOI: 10.1145/3375627.3375850. URL: http://dx.doi.org/10.1145/3375627.3375850.

[12] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. “"Why Should I Trust You?": Explaining the Predictions of Any Classifier”. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*. 2016, pp. 1135–1144.

[13] Arnaud Van Looveren and Janis Klaise. “Interpretable Counterfactual Explanations Guided by Prototypes”. In: *Age* 46 (), p. 46.

[14] Sandra Wachter, Brent Mittelstadt, and Chris Russell. “Counterfactual explanations without opening the black box: Automated decisions and the GDPR”. In: *Harv. JL & Tech.* 31 (2017), p. 841.

A Simple Pyramid Vision Transformer for Human Pose Estimation in Crowds

Mickael Cormier

Vision and Fusion Laboratory
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
mickael.cormier@kit.edu

Abstract

Multi-person Pose Estimation is essential for several computer vision tasks related to motion analysis and anomaly detection. The impressive and continual progress in this field leads to application in uncooperative real-world scenarios such as detecting anomalous and dangerous behavior from individuals or groups within dense crowds in public places. However, reliably detecting poses within crowds in surveillance footage remains a very challenging task, due to diverse occlusions, illumination changes and long processing time. In this work, we present a simple Pyramid Vision Transformer for Human Pose Estimation achieving competitive results on the COCO Keypoints 2017 [16] while requiring significantly less parameters and thus computation time. A significant improvement is reported over the baselines on the more crowded OCHuman [33], PoseTrack 2018 [2], and CrowdPose [14] datasets.

1 Introduction

Human Pose Estimation (HPE) is a computer vision task which has made impressive progress over the last few years [5, 13, 8, 27, 30, 31, 15]. Applications

include pedestrian gait recognition [26] and more generally action recognition [11]. While HPE in controlled environment delivers convincing results, multiple challenges arise for application in real-world uncontrolled scenarios, such as computation time for larger crowds, elevated view on persons, partial or almost total occlusion by diverse buildings of infrastructures, other persons or even self-occlusion [10].

In this work, we leverage the power of emerging Transformer architectures and based a on several best practices and experiments we propose a simple yet efficient model to reduce computation time while improving results, especially on occluded detections.

2 Related work

Mutiple datasets for HPE have been released in the last years [16, 1, 2, 14, 33]. One the most popular, the COCO Keypoints 2017 [16], offers over 200,000 images and 250,000 poses in single images with common poses and a frontal view. PoseTrack18 [2] features video frames with more complex real life scenarios in controlled environments, such as sport events, and is based on the MPII dataset [1]. Smaller datasets such as OCHuman [33] and CrowdPose [14] specifically address (self-)occlusion with similar frontal views on single images with two subjects for the former, and crowds in controlled environments such as group photos or sport events for the latter.

Different topologies of the human pose are proposed with different number of keypoints. In COCO a human pose is represented by 17 keypoints, of which five (nose, eyes, ears) are on the head. The MPII and Posetrack18 topologies simplify the pose by reducing the head keypoints to two and three, respectively, i.e. Posetrack18 has three keypoints for the head: top of head, nose and neck.

Two main categories of approaches have been presented in recent years to tackle HPE. First, bottom-up methods [5, 13, 8] detect all body parts in an image and fuse the retrieved keypoints to create a human pose. Since these methods detect keypoints independently from the actual person count on the image, the inference time is independent of the amount of people present. Second, top-down methods composed of a person detector and a pose estimator predict the bounding

boxes and poses separately. Recently Transformer-based approaches [31, 15] challenged the mainly CNN [27, 30] dominated field. The quality of a top-down method is, however, highly dependent on the quality of the person detection and the inference time increases relatively to the person count. In this work we attempt to reduce this inference time significantly while improving the quality of the predictions.

3 Methods

Vision Transformers have been recently applied to HPE with impressive results and significantly less parameters [31, 15]. The work propose by Yang et al. [31] proposed an hybrid model which relies on a shorted CNN model and a Zransformer head. Following this work, we propose a simple and flexible full Transformer model, compose of three main components: a Transformer-based backbone instead of a CNN for feature extraction, a Transformer Encoder to model long-range relationship between feature vectors, and a head for keypoints heatmaps prediction. The architecture is illustrated in Figure 3.1. In the reminder of this section, we described each part.

3.1 Transformer Backbone

While hybrid architectures combining CNNs and Transformers have shown impressive results, recent works have shown that Transformer-based backbones improve performance on several vision tasks [28, 17] and seem more robust to severe occlusions, perturbations, and domain shifts [18]. We argue that these properties are beneficial to HPE and therefore, we adopt the recent Pyramid Vision Transformer (PVT)v2 [28] designed for pixel-level dense prediction tasks as our backbone. Following the idea of shortening the backbone from [31], we chose to reduce the original four stages from our backbones to only two stages.

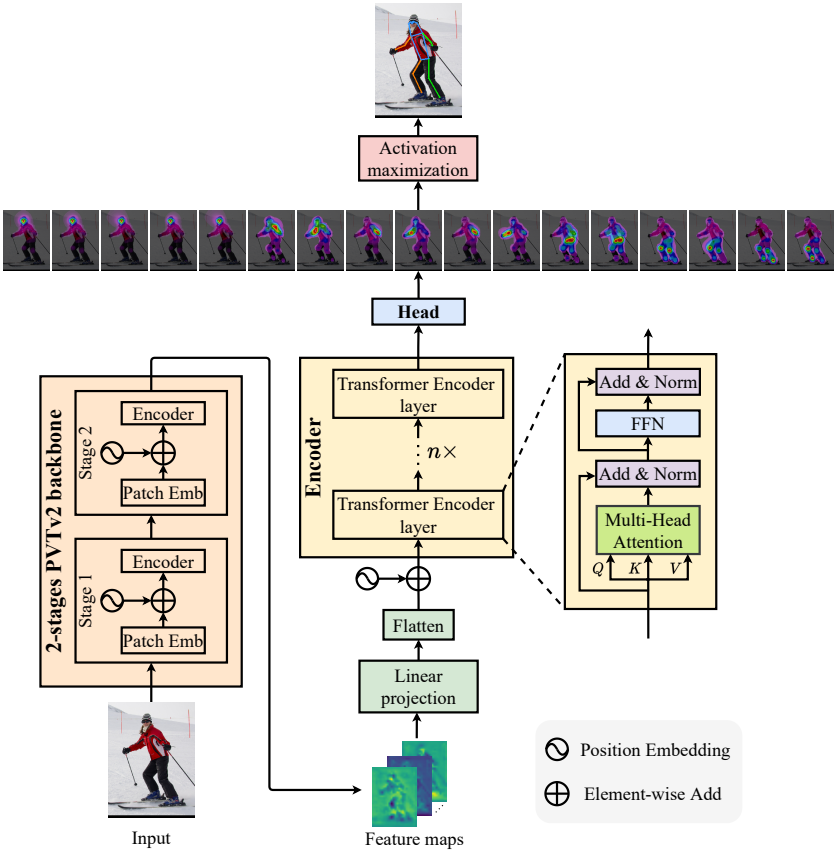


Figure 3.1: Overview of our model architecture. For an input detection image, a shortened 2-stage PVTv2 [28] backbone extracts feature maps, which are then flattened into fixed-size feature vectors and added with position embeddings. Subsequently, the dependencies between feature vectors in sequence are modeled by Transformer Encoder layers. Finally, a lightweight head is attached to predict the keypoint heatmaps.

3.2 Transformer Encoder

Following [31], we choose to encode the long-range relationships between the rich features with a Transformer Encoder.

Given an input image $\mathbf{I} \in \mathbb{R}^{3\times H_I \times W_I}$, the backbone extracts the low-level features and outputs feature maps $\mathbf{X}_f$ of size $d_f \times H \times W$, in this case, $(H, W) = (H_I/8, W_I/8)$. The feature maps are then linearly embedded and their dimension is transformed to d. The transformed feature maps are finally flattened into a 1D fixed-size feature vector sequence $\mathbf{X} \in \mathbb{R}^{L\times d}$, where $L = H \cdot W$. To retain positional information, a fixed 2D position encoding $\mathbf{E}_{\text{pos}}$ is added to the sequence as proposed in recent works [21, 6, 31, 15]. To retain positional information, a fixed 2D position encoding $\mathbf{E}_{\text{pos}}$ [31, 15] is added to the sequence (Eq. (3.1)).

$$\mathbf{Z}_0 = \mathbf{X} + \mathbf{E}_{\text{pos}} \tag{3.1}$$

Subsequently, $\mathbf{Z}_0$ enters the Transformer Encoder, which consists of n Transformer Encoder layers. Concretely, each Transformer Encoder layer comprises a Multi-head Self-Attention (MSA) sub-layer (Eq. (3.2)) and a feed-forward network (FFN) sub-layer (Eq. (3.3)). The FFN contains two linear transformations with a ReLU non-linearity in between. Moreover, residual connection followed by LayerNorm (LN) [3] is applied around each of the two sub-layers.

$$\mathbf{Z}'_i = \mathrm{LN}(\mathrm{MSA}(\mathbf{Z}_{i-1}) + \mathbf{Z}_{i-1}), \qquad i = 1, \ldots, n \tag{3.2}$$

$$\mathbf{Z}_i = \mathrm{LN}(\mathrm{FFN}(\mathbf{Z}'_i) + \mathbf{Z}'_i), \qquad i = 1, \ldots, n \tag{3.3}$$

3.3 Regression Head

Heatmaps predictions are obtained for each keypoint by a regression head following the output of the Encoder. For an input sequence $\mathbf{X} \in \mathbb{R}^{L\times d}$, the Encoder outputs a sequence $\mathbf{E} \in \mathbb{R}^{L\times d}$. The output $\mathbf{E}$ is then reshaped back to the shape of $d \times H \times W$, where here $(H, W) = (H_I/8, W_I/8)$. Following common practice [30, 25, 31, 15], the resolution of the heatmap is set to a quarter of the input image, *i.e.* $(H', W') = (H_I/4, W_I/4)$. Hence, a deconvolution

layer is added for upsampling [30, 8]. Finally, for the heatmaps prediction $\mathbf{H} \in \mathbf{R}^{K \times H' \times W'}$ of K different keypoints, the channel dimension of $\mathbf{E}$ is reduced from d to K via 1×1 convolution.

3.4 Model Variants

Since the Pyramid Vision Transformerv2 backbone is originally proposed in different scales, we also use seven backbone variations for our model, as described in Table 3.1. We follow the model naming convention in [31]. The name of our model is composed of three parts: the prefix "TP", the name of the backbone, and the number of Transformer Encoder layers. For instance a model called "TP-P-B0-A4" is composed of a Pyramid Vision Transformerv2-B0 backbone (abbreviated as P-B0) and a Transformer Encoder containing 4 Encoder layers.

All model variants produce 128 channels feature maps except for PVTv2-B0 with 64 channels. Following [28], the resolution of the feature map is always 1/8 of the input image. For the Transformer Encoder of all variants, we simply follow the setting of TransPose-R [31]. The dimension of the Transformer Encoder is $d = 256$. We employ $n = 4$ Transformer Encoder layers in total. In each layer, the number of heads for MSA and the number of hidden units for FFN is set to 8 and $h = 1024$, respectively. In the head, the upsampling is achieved via a 4×4 deconvolution.

Model Name	**Backbone**			**Transformer Encoder**				**Head**	
	Backbone	d_f	**Downsampling**	**#Encoder layers**	**#Heads**	d	h	**#DECONV layers**	**Kernel size**
TP-P-B0-A4	PVTv2-B0*	64	1/8	4	8	256	1024	1	4
TP-P-B1-A4	PVTv2-B1*	128	1/8	4	8	256	1024	1	4
TP-P-B2-A4	PVTv2-B2*	128	1/8	4	8	256	1024	1	4
TP-P-B2_Li-A4	PVTv2-B2_Li*	128	1/8	4	8	256	1024	1	4
TP-P-B3-A4	PVTv2-B3*	128	1/8	4	8	256	1024	1	4
TP-P-B4-A4	PVTv2-B4*	128	1/8	4	8	256	1024	1	4
TP-P-B5-A4	PVTv2-B5*	128	1/8	4	8	256	1024	1	4

Table 3.1: Architecture configuration details of different variants of our propose model. The star symbol (*) indicates that the Pyramid Vision Transformerv2 backbone [28] is reduced from the original four stages to two. d_f, d, and h are the dimension of feature maps, the dimension of Transformer Encoder, and the dimension of hidden layer in FFN of Transformer Encoder layer.

4 Evaluation

We first evaluate different variations of our models regarding architectural choices. We then quantitatively evaluate our models on four different datasets.

4.1 Ablation Study

We perform ablation studies on the COCO [16] dataset. As in [30, 25, 31, 15], we first extend the ground truth human bounding boxe to a fixed aspect ratio $height : width = 4 : 3$. Then we crop and resize the bounding boxe from the original image to a fixed size 256×192. To reduce overfitting, we apply standard data augmentation techniques, including random scale ($\pm 30\%$), random rotation ($[-45°, 45°]$), and flipping. We also use half body augmentation [29]. The reduced Pyramid Vision Transformerv2 [28] backbone network is initialized with the weights pre-trained on ImageNet-1K classification task [24].

All models are trained for 230 epochs on two NVIDIA GeForce RTX 2080 Ti GPUs using Adam [12] as optimizer and a cosine annealing learning rate schedule from $2e-4$ to $2e-5$.

4.1.1 Number of Backbone Stages

We analyze the number of stages of Pyramid Vision Transformerv2 [28] backbone using TP-P-B0-A4 with originally 4 stages. We reduce to 3 and 2 stages and report the results in Table 4.1. As suggested in [31], reducing the number of stages yields better results with fewer parameters, e.g. the model with 2 stages backbone achieves the best AP. We also observe the similar tendency for other variants, as shown in Figure 4.1.

4.1.2 Number of Transformer Encoder Layers

We then evaluate the influence of the number of Transformer layers in the encoder on the performance of the model. To this aim, TP-P-B0 and TP-P-B2_Li are used which represent models with small and medium size respectively. For

Model	Backbone	#Stages	AP ↑	Params (M) ↓
TP-P-B0-A4	Pyramid Vision Transformerv2-B0 [28]	2	**74.2**	**4.74**
		3	74.0	6.74
		4	72.6	9.79

Table 4.1: Ablation study on number of stages of Pyramid Vision Transformerv2 [28] backbone on COCO [16] validation set with ground truth human bounding boxes. ↑/↓ indicates that the higher/lower, the better. The best value in each column is marked in bold.

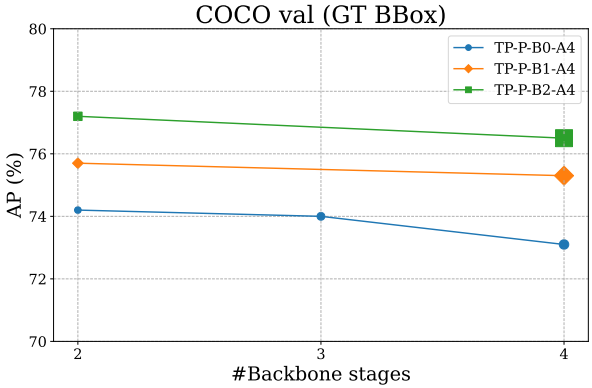


Figure 4.1: Effect of number of stages of Pyramid Vision Transformerv2 [28] backbone. AP is measured on COCO [16] validation set with ground truth human bounding boxes. Model size is indicated by marker size. For all models, AP drops with increasing number of backbone stages.

Model	#Layers	AP ↑	$AP_{0.5}$ ↑	$AP_{0.75}$ ↑	AP_M ↑	AP_L ↑	AR ↑	Params (M) ↓
TP-P-B0	2	69.1	90.3	76.3	66.0	73.5	72.2	**3.1**
	4	74.2	92.5	81.6	71.1	78.8	76.8	4.7
	6	**75.7**	**92.6**	**82.7**	**72.7**	**80.4**	**78.5**	6.3
TP-P-B2_Li	2	75.6	92.5	82.5	72.7	80.3	78.4	**4.5**
	4	77.1	**93.5**	**84.8**	74.2	81.7	79.8	6.0
	6	**77.6**	**93.5**	84.7	**74.6**	**82.2**	**80.2**	7.6

Table 4.2: Ablation study on Transformer Encoder size on COCO [16] validation set with ground truth human bounding boxes. "#Layers" refers to the number of Transformer Encoder layers. ↑/↓ indicates that the higher/lower, the better. For each model, the best value in each column is marked in bold.

a fair comparison, all models are trained on COCO [16] dataset using the same configuration and strategy. Results are evaluated on the COCO validation set with ground truth human bounding boxes and summarized in Table 4.2.

For both models the overall performance indicated by AP improves accordingly as the number of Transformer Encoder layers increases. We observe in more depth that the AP_M (AP for medium objects) and AP_L (AP for large objects) benefit largely from more Transformer layers. For example, when increasing the number of Encoder layers of TP-P-B0 from two to four, AP_M climbs rapidly ($+5.1$).

In addition, the impact of scaling size of Transformer Encoder varies for backbones of different sizes, as compared in Table 4.2. For TP-P-B0, whose backbone is relatively small (0.5M), enlarging the Transformer Encoder from 2 layers to 4 layers leads to a noticeable performance improvement ($+5.1$AP). In contrast, it only brings about a slight enhancement for TP-P-B2_Li which is a larger backbone (1.8M). These results seem to concur with similar experiments in [31], in which ResNet based models require more encoder layers than HrNet based models. Therefore, there is a clear trade-off between backbone and encoder layers. While the backbone is usually responsible for large part of the inference time, increasing the number of Transformer layers in the encoder seem to compensate to some extent in term of quality.

4.2 Quantitative Results

We further conduct extensive performance studies on four popular datasets with different topologies and challenges, described in Table 4.2.

4.2.1 COCO

Unless otherwise specified, the models are trained using the same settings as mentioned in Section 4.1. Similar as [30, 25, 7], we apply a two-stage top-down paradigm. We use the same person detection result as in [30, 25], which is generated by an off-the-shell faster-RCNN detector [23] with person detection AP 56.4 on COCO validation set. Following the common practice [30, 25, 19,

Dataset	#Images	#Labeled Person	#Keypoints
COCO Keypoints 2017 [16]	$>$ 200,000	$>$ 250,000	17
PoseTrack 2018 [2]	66,374	153,615	15
OCHuman [33]	4731	8110	17
CrowdPose [14]	20,000	80,000	14

Table 4.3: Comparison between COCO Keypoints [16], PoseTrack [2], OCHuman [33], and CrowdPose [14] in terms of number of images, number of labeled person instances, and number of keypoints annotation of an individual person.

Model	AP ↑	$AP_{0.5}$ ↑	$AP_{0.75}$ ↑	AP_M ↑	AP_L ↑	AR ↑	Params (M) ↓	FPS ↑
TP-R-A4 [31]	75.1	92.6	82.6	71.9	79.6	77.8	5.8	**135.9**
TP-H-A6 [31]	**78.1**	**93.6**	84.6	**74.9**	**82.6**	**80.5**	17.2	32.3
TP-P-B0-A4	74.2	92.5	81.6	71.1	78.8	76.8	**4.7**	126.3
TP-P-B1-A4	75.7	92.5	82.7	72.6	80.6	78.6	6.2	120.9
TP-P-B2-A4	77.2	93.5	**84.8**	74.1	82.0	79.8	7.8	91.0
TP-P-B2_Li-A4	77.1	93.5	**84.8**	74.2	81.7	79.8	6.0	88.2
TP-P-B3-A4	77.1	**93.6**	83.8	74.2	81.8	79.8	7.8	91.8
TP-P-B4-A4	77.7	93.5	84.7	74.6	82.3	80.3	10.2	67.4
TP-P-B5-A4	76.7	93.5	82.8	73.5	81.4	79.3	8.1	78.3

Table 4.4: Results on COCO [16] validation set with ground truth bounding boxes. The input size is 256×192. ↑/↓ indicates that the higher/lower, the better. The best value in each column is marked in bold.

Model	AP ↑	$AP_{0.5}$ ↑	$AP_{0.75}$ ↑	AP_M ↑	AP_L ↑	AR ↑	Params (M) ↓	FPS ↑
TP-R-A4 [31]	72.6	89.1	79.9	68.8	79.8	78.0	5.8	**135.9**
TP-H-A6 [31]	**75.8**	**90.1**	**82.1**	**71.9**	**82.8**	**80.8**	17.2	32.3
TP-P-B0-A4	71.9	88.9	79.0	68.2	78.9	77.2	**4.7**	126.3
TP-P-B1-A4	73.6	89.6	80.3	69.9	80.6	78.7	6.2	120.9
TP-P-B2-A4	74.4	89.7	81.2	70.7	81.6	79.6	7.8	91.0
TP-P-B2_Li-A4	74.7	89.8	81.5	71.0	81.6	79.7	6.0	88.2
TP-P-B3-A4	74.8	90.0	81.5	71.0	81.8	79.9	7.8	91.8
TP-P-B4-A4	75.2	89.9	**82.1**	71.2	82.4	80.3	10.2	67.4
TP-P-B5-A4	74.2	89.7	80.7	70.4	81.4	79.5	8.1	78.3

Table 4.5: Results on COCO [16] validation set with detected human boxes generated by faster-RCNN [23] detector having human AP of 56.4. The input size is 256×192. ↑/↓ indicates that the higher/lower, the better. The best value in each column is marked in bold.

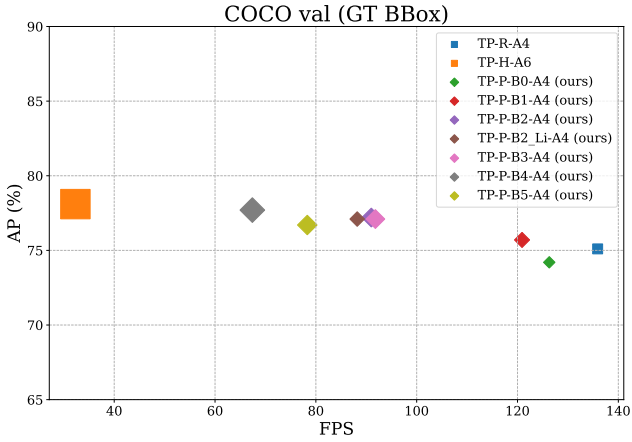
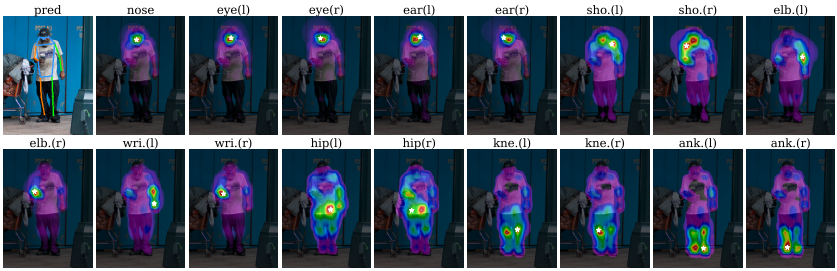


Figure 4.2: Model comparison on COCO [16] validation set with ground truth bounding boxe in aspects of model size, accuracy, and speed. □ and ◊ correspond to TransPose models [31] and our proposed models, respectively. Larger symbol indicates model with larger number of parameters.

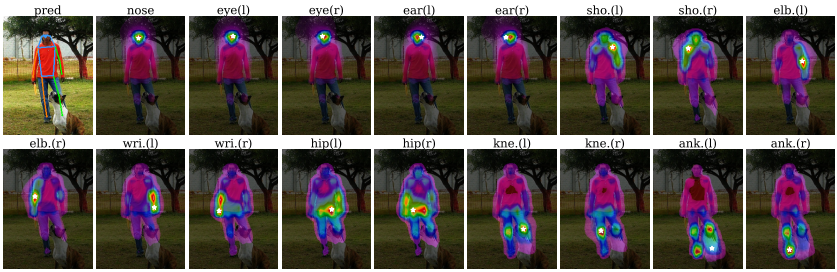
7] to generate final heatmap prediction, we run the input image as well as its horizontally flipped version through the network and average the results. To alleviate error when decoding the predicted downscaled heatmaps into the final joint coordinates in the original image, we adopt Distribution-Aware coordinate Representation of Keypoint (DARK) [32] and its decoding strategy. Pose rescoring strategy and OKS-based non maximal suppression (NMS) [20] are also employed.

Finally, we visualize the position prediction and attention maps for different keypoints in Figure 4.3, for a single person with and without occlusion. While the non-occluded case seems straightforward, we observe that the model learns context information, especially for the shoulders, hips, knees and ankles. In the occluded case, the left knee and left ankle are occluded by a dog in the front. The model is still able to predict the accurate location using the context. For the partly occluded left knee, the model is able to pay attention to the relatively accurate area, with the help of the symmetrical joint (right knee). For the completely occluded left ankle, the attention focuses mainly to its nearby joints

on the same side (left knee) and its symmetrical joint (right ankle). Based on these spatial clues, the model predicts the possible location where the left ankle is probably located.



(a) A single person standing without occlusion.



(b) A single person standing with occlusion.

Figure 4.3: Comparison of attention maps of the last Transformer Encoder layer between a single person standing *without* occlusion and a single person standing *with* occlusion. In each subfigure, the top left image is the input image annotated with the predicted pose. Pose prediction and attention maps are generated by TP-P-B2_Li-A4.

4.2.2 OCHuman

Following the setting from [33], we first train all models on COCO as in Section 4.1. The robustness of our models against strong occlusion is validated

on the validation and test set of OCHuman with ground truth bounding boxes. The TransPose [31] models are tested and compared to as a baseline. We report the results on the validation set and test set in Table 4.6 and Table 4.7, respectively.

Model	AP ↑	$AP_{0.5}$ ↑	$AP_{0.75}$ ↑	AP_L ↑	AR ↑	Params (M) ↓	FPS ↑
TP-R-A4 [31]	62.0	80.3	66.7	62	66.2	5.8	**135.9**
TP-H-A6 [31]	62.3	77.2	67.9	62.4	66.6	17.2	32.3
TP-P-B0-A4	60.9	81.4	66.4	60.9	65.4	**4.7**	126.3
TP-P-B1-A4	62.8	81.6	68.1	62.8	66.7	6.2	120.9
TP-P-B2-A4	65.0	**81.9**	70.4	65.0	68.8	7.8	91.0
TP-P-B2_Li-A4	64.7	81.7	70.1	64.7	68.8	6.0	88.2
TP-P-B3-A4	65.0	81.7	70.1	65.0	68.9	7.8	91.8
TP-P-B4-A4	**65.9**	81.8	**71.4**	**65.9**	**69.5**	10.2	67.4
TP-P-B5-A4	65.0	81.7	71.2	65.0	69.1	8.1	78.3

Table 4.6: Results on OCHuman [33] validation set with ground truth bounding boxes. The input size is 256×192. ↑/↓ indicates that the higher/lower, the better. The best value in each column is marked in bold.

Model	AP ↑	$AP_{0.5}$ ↑	$AP_{0.75}$ ↑	AP_L ↑	AR ↑	Params (M) ↓	FPS ↑
TP-R-A4 [31]	61.8	78.5	67.2	61.8	65.9	5.8	**135.9**
TP-H-A6 [31]	62.0	76.6	66.9	62.1	66.3	17.2	32.3
TP-P-B0-A4	61.2	80.4	67.0	61.2	65.3	**4.7**	126.3
TP-P-B1-A4	63.0	80.6	68.4	63.0	66.8	6.2	120.9
TP-P-B2-A4	65.0	**81.7**	**70.4**	65.0	68.9	7.8	91.0
TP-P-B2_Li-A4	65.1	**81.7**	70.4	65.1	**69.0**	6.0	88.2
TP-P-B3-A4	64.6	81.4	69.3	64.6	68.4	7.8	91.8
TP-P-B4-A4	**65.2**	80.4	70.3	**65.2**	68.8	10.2	67.4
TP-P-B5-A4	64.8	81.6	**70.4**	64.8	68.7	8.1	78.3

Table 4.7: Results on OCHuman [33] test set with ground truth bounding boxes. The input size is 256×192. ↑/↓ indicates that the higher/lower, the better. The best value in each column is marked in bold.

As stated earlier, one of our motivation for replacing the CNN backbone with a Transformer backbone is to improve the robustness of our model against occlusion. This assumption is here largely proven. While our models are beaten

by the HrNet backbone in TP-H-A6 [31] on the COCO dataset, our models surpass it largely when conducting evaluation on the largely occluded OCHuman dataset. The best-performing model TP-P-B4-A4 greatly outperforms TP-H-A6 on the validation set (+3.6AP) and on the test set (+3.2AP) with much fewer parameters and twice its speed. Moreover, almost all variants of our model achieve better performance than TP-R-A4 and TP-H-A6 on both validation set and test set. This is mainly due to more than 30% persons in OCHuman being under heavy occlusion (MaxIoU > 0.75), compared to less than 0.1% for COCO [33].

4.2.3 PoseTrack18

Model	Head AP ↑	Shou AP ↑	Elb AP ↑	Wri AP ↑	Hip AP ↑	Knee AP ↑	Ankl AP ↑	Total AP ↑
TP-R-A4 [31]	86.8	88.9	83.9	78.2	82.3	81.8	78.0	83.1
TP-H-A6 [31]	87.0	89.3	84.8	79.6	82.5	82.7	78.9	82.8
TP-P-B0-A4	86.5	88.4	83.0	76.8	81.0	80.6	76.9	82.2
TP-P-B1-A4	87.2	89.9	84.1	78.7	81.8	81.6	78.5	83.3
TP-P-B2-A4	86.9	89.2	85.1	80.0	82.5	82.6	79.5	83.9
TP-P-B2_Li-A4	87.5	**90.6**	85.6	80.5	82.1	83.3	79.7	84.3
TP-P-B3-A4	87.5	**90.6**	85.5	80.4	82.5	83.6	80.5	84.4
TP-P-B4-A4	**88.1**	90.2	**85.7**	**81.3**	**82.9**	**84.2**	**81.0**	**85.0**
TP-P-B5-A4	87.3	89.7	85.5	80.3	82.0	83.0	79.7	84.2

Table 4.8: Results on PoseTrack18 [2] validation set with ground truth bounding boxes. The input size is 256×192. ↑/↓ indicates that the higher/lower, the better. The best value in each column is marked in bold.

We further focus on the PoseTrack18 dataset [2] and its multi person pose estimation task with a topology of 15 keypoints. To this aim, we reuse our models pre-trained on COCO. The training setup as well as the data augmentation are almost the same as those for COCO, described in Section 4.1. We start with re-initializing the final layer uniformly. We train only the new final layer with initial learning rate $1e-4$ for 30 epochs, while freezing other parts of the model. Finally, we finetune the entire model for another 30 epochs using a smaller starting learning rate ($5e-5$). The cosine annealing learning rate scheduler is involved in both steps. For testing we adopt the person detection results provided by `mmpose` [9], which are generated by a Cascade R-CNN

(X-101-64x4d-FPN) [4] human detector. Other testing configurations remain the same as for COCO [16]. We report the results on the validation set with ground truth bounding boxes in Table 4.8 and report not only AP but also APs of different keypoints. Most of our models surpass the baseline models, due to main AP gain from the wrists and knees, which are more volatile joints at the far ends, often subjects to occlusions.

4.2.4 CrowdPose

Model	AP	$AP_{0.5}$ ↑	$AP_{0.75}$ ↑	AR ↑	AP_E ↑	AP_M ↑	AP_H ↑	Params (M) ↓	FPS ↑
TP-R-A4 [31]	69.8	83.7	75.7	78.8	79.7	71.2	56.4	5.8	**135.9**
TP-H-A6 [31]	71.3	83.6	76.5	80.3	80.5	72.7	58.3	17.2	32.3
TP-P-B0-A4	68.2	83.1	73.7	77.5	78.4	69.5	54.2	**4.7**	126.3
TP-P-B1-A4	70.3	83.6	75.8	79.5	80.0	71.8	56.7	6.2	120.9
TP-P-B2-A4	71.7	83.8	76.9	80.8	81.3	73.2	58.0	7.8	91.0
TP-P-B2_Li-A4	71.7	83.7	76.9	81.0	81.3	73.3	58.3	6.0	88.2
TP-P-B3-A4	71.8	83.7	76.9	81.0	81.3	73.4	58.2	7.8	91.8
TP-P-B4-A4	**72.7**	**84.2**	**77.6**	**82.1**	**81.8**	**74.3**	**59.2**	10.2	67.4
TP-P-B5-A4	71.4	83.6	76.7	80.6	81.0	73.0	57.6	8.1	78.3

Table 4.9: Results on CrowdPose [14] test set with detected human boxes generated by YOLOv3 [22] detector. The input size is 256×192. ↑/↓ indicates that the higher/lower, the better. The best value in each column is marked in bold. E, M, and H of AP stand for crowding levels easy, medium, and hard, as defined in [14].

For the CrowdPose dataset [14], we use the same two-stage finetuning strategy as for PoseTrack18 (see Section 4.2.3). All models are evaluated on CrowdPose test set with detected human bounding boxes generated by a YOLOv3 detector. Our results are reported in Table 4.9. We also report the results on three crowding levels, *i.e.*, uncrowded (easy), medium crowded, and extremely crowded (hard), as defined in [14]. Almost all of our proposed models surpass the baselines, with the exception of two. Our best-performing model TP-P-B4-A4, outperforms TP-H-A6 by a large margin of $+1.4$ AP with significantly fewer parameters and more than twice its speed. The strongest difference comes from AP_E ($+1.3$AP) and AP_M ($+1.6$AP) while AP_H is also moderately improved ($+0.9$AP). The results demonstrate that our models are able to handle not only simple daily but also crowded cases, probably due to better performance in occluded cases.

5 Conclusion

We engineer a full Transformer-based model for top-down HPE. The recipe is simple, flexible and can be applied with several variations: a reduced Transformer-based backbone without convolutions for feature extraction, a Transformer Encoder to model long-range relationship between feature vectors, and a simple head for keypoint heatmap estimation. Our results show that our model performs competitively and even outperforms the much heavier baseline on three out of four datasets, with heavy occlusions and higher levels of crowdedness. Future works should focus on difficult occlusions for which multiple person are visible in a detected bounding boxe often leading to predictions of the right keypoint belonging to the wrong person.

Acknowledgments

This work is based on joint works resulting from a very close collaboration between Mickael Cormier with his Master Thesis student Haobin Tan. Both the author and the corresponding student have contributed substantially to this research. While it is difficult to set a precise boundary, Mickael Cormier was rather in charge of the idea, while the student focused on the implementation.

References

[1] Mykhaylo Andriluka et al. “2D Human Pose Estimation: New Benchmark and State of the Art Analysis”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2014.

[2] Mykhaylo Andriluka et al. “Posetrack: A benchmark for human pose estimation and tracking”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 5167–5176.

[3] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. “Layer normalization”. In: *arXiv preprint arXiv:1607.06450* (2016).

[4] Zhaowei Cai and Nuno Vasconcelos. “Cascade r-cnn: Delving into high quality object detection”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 6154–6162.

[5] Z. Cao et al. “OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2019).

[6] Nicolas Carion et al. *End-to-End Object Detection with Transformers*. 2020. arXiv: 2005.12872 [cs.CV].

[7] Yilun Chen et al. “Cascaded pyramid network for multi-person pose estimation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 7103–7112.

[8] Bowen Cheng et al. “Higherhrnet: Scale-aware representation learning for bottom-up human pose estimation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 5386–5395.

[9] MMPose Contributors. *OpenMMLab Pose Estimation Toolbox and Benchmark*. https://github.com/open-mmlab/mmpose. 2020.

[10] Mickael Cormier et al. “Where Are We With Human Pose Estimation in Real-World Surveillance?” In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2022, pp. 591–601.

[11] Haodong Duan et al. “Revisiting Skeleton-based Action Recognition”. In: *arXiv preprint arXiv:2104.13586* (2021).

[12] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: 1412.6980 [cs.LG].

[13] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. “Pifpaf: Composite fields for human pose estimation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 11977–11986.

[14] Jiefeng Li et al. *CrowdPose: Efficient Crowded Scenes Pose Estimation and A New Benchmark*. 2019. arXiv: 1812.00324 [cs.CV].

[15] Yanjie Li et al. *TokenPose: Learning Keypoint Tokens for Human Pose Estimation*. 2021. arXiv: 2104.03516 [cs.CV].

[16] Tsung-Yi Lin et al. “Microsoft coco: Common objects in context”. In: *European conference on computer vision*. Springer. 2014, pp. 740–755.

[17] Ze Liu et al. *Swin Transformer: Hierarchical Vision Transformer using Shifted Windows*. 2021. arXiv: 2103.14030 [cs.CV].

[18] Muzammal Naseer et al. *Intriguing Properties of Vision Transformers*. 2021. arXiv: 2105.10497 [cs.CV].

[19] Alejandro Newell, Kaiyu Yang, and Jia Deng. “Stacked hourglass networks for human pose estimation”. In: *European conference on computer vision*. Springer. 2016, pp. 483–499.

[20] George Papandreou et al. “Towards accurate multi-person pose estimation in the wild”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 4903–4911.

[21] Niki Parmar et al. *Image Transformer*. 2018. arXiv: 1802.05751 [cs.CV].

[22] Joseph Redmon and Ali Farhadi. *YOLOv3: An Incremental Improvement*. 2018. arXiv: 1804.02767 [cs.CV].

[23] Shaoqing Ren et al. “Faster r-cnn: Towards real-time object detection with region proposal networks”. In: *Advances in neural information processing systems* 28 (2015), pp. 91–99.

[24] Olga Russakovsky et al. “Imagenet large scale visual recognition challenge”. In: *International journal of computer vision* 115.3 (2015), pp. 211–252.

[25] Ke Sun et al. “Deep high-resolution representation learning for human pose estimation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 5693–5703.

[26] Torben Teepe et al. “GaitGraph: Graph Convolutional Network for Skeleton-Based Gait Recognition”. In: *arXiv preprint arXiv:2101.11228* (2021).

[27] Jingdong Wang et al. “Deep high-resolution representation learning for visual recognition”. In: *IEEE transactions on pattern analysis and machine intelligence* (2020).

[28] Wenhai Wang et al. *PVTv2: Improved Baselines with Pyramid Vision Transformer*. 2021. arXiv: 2106.13797 [cs.CV].

[29] Zhicheng Wang et al. "Mscoco keypoints challenge 2018". In: *Joint Recognition Challenge Workshop at ECCV 2018*. Vol. 5. 2018.

[30] Bin Xiao, Haiping Wu, and Yichen Wei. "Simple baselines for human pose estimation and tracking". In: *Proceedings of the European conference on computer vision (ECCV)*. 2018, pp. 466–481.

[31] Sen Yang et al. "TransPose: Keypoint localization via transformer". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 11802–11812.

[32] Feng Zhang et al. "Distribution-aware coordinate representation for human pose estimation". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 7093–7102.

[33] Song-Hai Zhang et al. *Pose2Seg: Detection Free Human Instance Segmentation*. 2019. arXiv: 1803.10683 [cs.CV].

Temporal Bird's Eye View for 3D Semantic Segmentation

Fabian Duerr

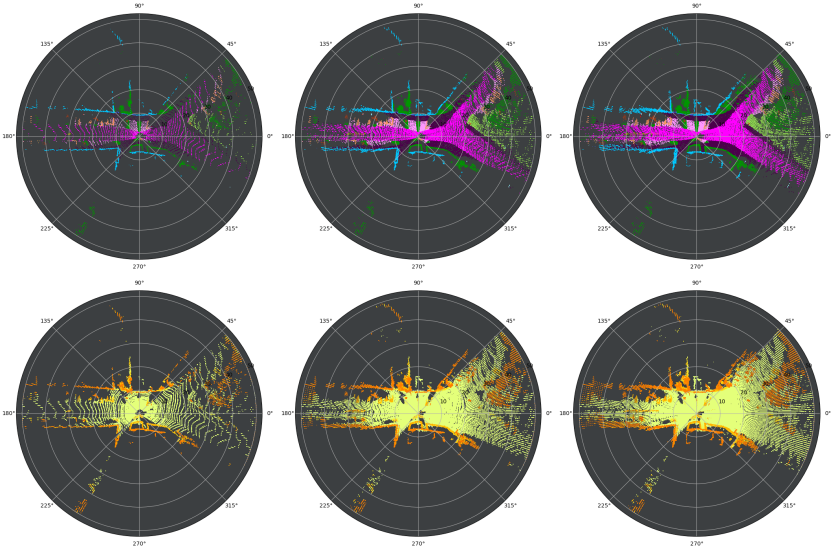
Vision and Fusion Laboratory
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
fabian.duerr@partner.kit.edu

Abstract

Due to the growing importance of autonomous robots and vehicles, 3D semantic segmentation, a key task of 3D scene understanding, has become more and more important. Despite its sequential nature in real-time scenarios, 3D semantic segmentation is often approached as single frame problem. However, temporal dependencies and information offer a huge potential to improve the predictions. Therefore, we propose a recurrent temporal architecture for 3D semantic segmentation, which exploits temporal information at the input and feature stage, to maximize the temporal benefits. Aggregated point clouds in bird's eye view increase the information provided to the backbone and temporally fused feature maps exploit temporal dependencies on feature level. The experiments conducted on a challenging and large-scale outdoor dataset show considerable improvements compared to a single frame baseline. The temporal information improve the results for every individual class.

1 Introduction

Living in a 3D world, one of the key challenges for autonomous robots is the understanding and interpretation of their 3D environment. While point clouds



(a) Single frame. (b) Five aggregated frames. (c) Ten aggregated frames.

Figure 1.1: Potential of aggregating consecutive frames in bird's eye view (BEV), visualized by the semantic segmentation at the top and the input point cloud colored by height at the bottom. While the single frame BEV (a) is relatively sparse, the aggregation of five (b) or ten previous frames (c) is much denser in the occupied areas and therefore adds a lot of information. This can not only be exploited for the shown input data but also for feature maps of the backbone.

already provide valuable geometric information, 3D semantic segmentation adds a class label to every individual point and therefore additional semantic information, which is often seen as key enabler for 3D scene understanding.
To tackle semantic segmentation of 3D point clouds, a proper representation or architecture is required to solve this task with established deep learning based approaches. While point-based approaches [21, 27] directly process raw point clouds, they deploy special architectures and convolution operations to deal with the unstructured data. To enable conventional convolutions and architectures, projection-based methods [20, 34] transform the point clouds into a regular space, e.g. grid.
An important property of real-time environment perception is the sequential

nature of the recorded sensor data. Temporal relations and information offer a huge potential to improve 3D semantic segmentation. As the environment does not change drastically during the recording of two consecutive frames, previous frames contain valuable information also for the current frame, see Fig. 1.1. The amount of information naturally diminishes with the temporal distance. For real-time applications, only past frames can be exploited whereas accessing future frames is not possible.
In this work, we present an efficient temporal semantic segmentation approach building upon bird's eye view (BEV) representation and exploiting temporal information at two stages. At the input stage, the point cloud of the current frame and the aggregated past point clouds are fused to increase the point cloud density and therefore input information. At feature stage, features of the current frame are fused with features from a temporal memory, which contains aggregated past information, following the idea of [10]. A feature alignment step in BEV space allows the reuse of computations from previous frames in both stages, which enables an efficient recurrent architecture. The benefits of the temporal fusion are twofold, it improves existing features by fusion, based on aggregated past information and additionally increases the density of the BEV.
To summarize our contributions, we propose:

- A temporal input memory, which efficiently aggregates input point clouds in BEV over time to increase information provided to the backbone.
- A temporal feature memory, which efficiently aggregates feature maps computed by the backbone, to provide aggregated information of the current frame and past frames to the semantic head, to further improve the predicted 3D semantic segmentation.

2 Related Work

2.1 3D Semantic Segmentation

As an integral part of 3D scene understanding, 3D semantic segmentation has drawn a lot of attention over the past years. Enabled by the availability of a

constantly increasing number of datasets [1, 3, 5, 29] considerable progress has been achieved. In contrast to images, a preliminary consideration about the representation is required, to tackle this task with Convolutional Neural Networks (CNN). Almost all representations proposed so far can be assigned to one of two main categories.
Point-based methods [14, 16, 21, 22, 27, 28] directly operate on the 3D point clouds and rely on adapted convolution operations and special network architectures. Projection-based methods transform the point clouds into a regular space, which enables the application of conventional convolutions and architectures. Based on the target space, these approaches can further be divided into subcategories, like dense and sparse voxel grids [7, 23, 26], range images [9, 20, 30] or bird's eye view (BEV). Zhang et al. [33] build upon a 3D occupancy grid but treat the z-axis as feature dimension and therefore work with a 2D BEV representation as input. PolarNet [34] proposes a 2D polar BEV representation, which is based on a learned PointNet [21] encoding of all points lying inside a BEV cell. In the last stage of the network the 2D feature maps are expanded to a 3D polar grid prediction. Motivated by the promising results of PolarNet and the general potential of the BEV representation for temporal fusion, the presented approach builds upon the polar BEV representation.

2.2 Temporal Point Cloud Fusion

The majority of the methods proposed so far treat semantic segmentation as single frame task. Most of the approaches, which exploit temporal information on feature level aim for 3D object detection [15, 17, 18, 24, 31], only a few approaches for 3D semantic segmentation exist [6, 8, 10, 25].
Yin et al. [31] tackle temporal object detection in BEV representation with an RNN-based architecture building upon an extended ConvGRU [2], called attentive spatio-temporal GRU. It aggregates spatio-temporal information to exploit temporal dependencies of the point cloud sequences. For the same task, Huang et al. [15] exploit a sparse 3D voxel representation and propose a LSTM to fuse sparse features from previous and the current frame. The object detection head is then applied to the temporally fused features.

For semantic segmentation, MinkowskiNet [8] uses the fourth dimension to include previous frames and relies on sparse convolutions to handle the dominating empty cells. One disadvantage is the dependence of the run-time on the number of past frames considered. SpSequenceNet [25], which relies on the backbone of [12], and a voxel-based representation, proposes a cross-frame global attention layer, which highlights features of the current frame based on past information. Cross-frame local interpolation targets the temporal combination of local information. However, this approach is only designed to exploit the last frame. Li et al. [6] exploit temporal information in range image space for moving object segmentation. Past distance information is transformed into the current frame and residual images are computed using the difference of the transformed past distance values and the current values. The residual images are then used as additional input channels to a CNN. TemporalLidarSeg [10] builds upon an RNN-based architecture and passes a recursively aggregated temporal memory through time. An alignment step improves temporal consistency by compensating the ego motion. The temporal information provided by the memory considerably improves the semantic segmentation results.
Most of the listed approaches are not capable of exploiting information over a larger number of past frames due to design [25] or increasing run-time [8, 31]. The presented approach however is able to exploit temporal information of sequences of arbitrary length in constant time, comparable to TemporalLidarSeg [10]. However, instead of using range images like the latter, the presented approach relies on the BEV representation, which offers additional possibilities for temporal fusion.

3 Recurrent 3D Semantic Segmentation

The goal of the presented approach is the exploitation of temporal information in BEV to improve 3D semantic segmentation. Therefore, the architecture is designed to use past information at two stages, see Fig. 3.1. Starting at the input stage, the BEV image fed to the backbone does not only contain the point cloud of the current time step but also the aggregated point clouds of the previous frames. The second temporal fusion stage is designed to fuse the feature maps

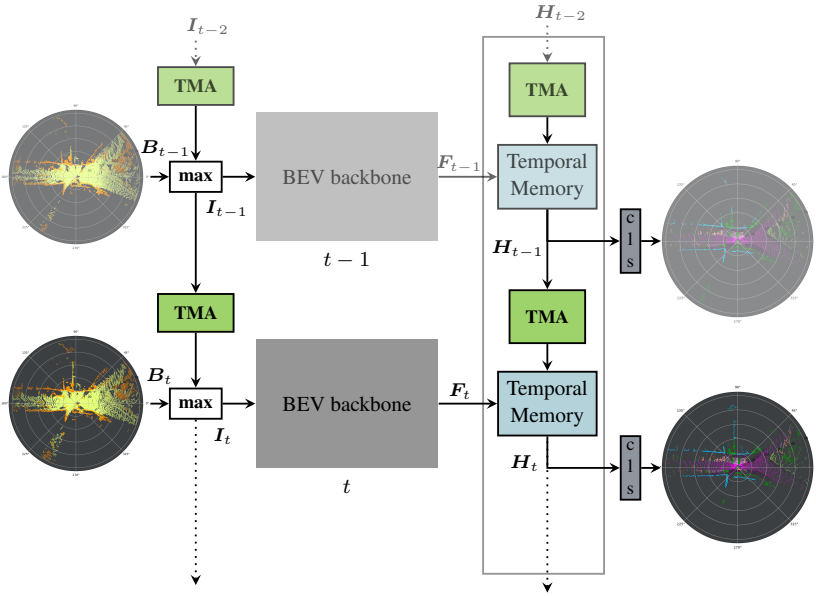


Figure 3.1: Overview of the recurrent temporal architecture, unrolled for two time steps. Aggregated input point clouds in BEV are fed to the backbone, which computes intermediate feature maps. Based on these features a temporal memory containing the aggregated past information is updated. The final semantic segmentation is computed from these temporally fused features.

computed by the BEV backbone. Therefore, they are used to update a temporal memory, which contains the aggregated past feature maps. The temporally fused and aggregated features are then expanded to a 3D polar grid and used for the final semantic predictions. Both stages efficiently reuse computations from the previous time steps.

Bird's Eye View Backbone Like single frame approaches, the presented architecture has a backbone responsible for computing feature maps of point clouds represented as BEV images. The differences however are twofold. First, while still taking only a single BEV image as input, it contains the aggregate point clouds of the current frame as well as past frames. Secondly, the intermediate

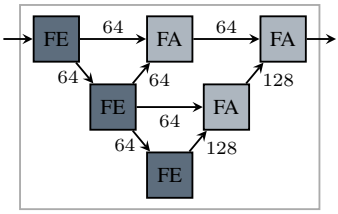


Figure 3.2: Backbone for computing feature maps for point clouds represented in BEV. The edges are labeled with the channel size of the feature maps.

feature maps are computed for the temporal memory instead of the semantic head, see Fig. 3.1.
In general, we build upon the ideas of PolarNet [34] but use a different backbone and reduce the channels of the PointNet, which encodes the polar input, to 16, 24 and 32. The deployed backbone is based on deep layer aggregation [32] and additionally motivated by LaserNet [19]. It is build of three feature extractors (FE) with four, five and six residual blocks [13] and a downsampling ratio of two. Feature aggregators (FA) apply a transposed convolution to upsample their lower resolution input and concatenate both inputs, followed by two residual blocks. The backbone architecture is depicted in Fig. 3.2.

Temporal BEV Alignment In order to realize the mentioned temporal aggregation of BEV images containing input point clouds or deep features, a recursive transformation of BEV images from the last to the current time step is required. This allows to reuse already aggregated point clouds or computed features of past frames, following the idea of [10]. The temporal alignment itself is required because of the ego motion, which changes the sensor origin.
Generally, an important relation for 3D semantic segmentation based on BEV images is the mapping from a 3D point $\boldsymbol{p} = (x, y, z)^T$ to its corresponding 2D BEV cell $\boldsymbol{u} = (u, v)^T$, which is described by

$$\boldsymbol{\mathcal{P}} : \mathbb{R}^3 \rightarrow [1, H] \times [1, W] \subset \mathbb{N}^2 \;\Rightarrow\; \boldsymbol{\mathcal{P}}(\boldsymbol{p}) = \begin{pmatrix} \left\lfloor \frac{\sqrt{x^2+y^2}}{\tilde{r}} \right\rfloor \\ \left\lfloor \frac{\text{atan2}(y,x)}{\tilde{\alpha}} \right\rfloor \end{pmatrix} = \begin{pmatrix} u \\ v \end{pmatrix}, \quad (3.1)$$

with the image resolution $(\tilde{r}, \tilde{\alpha})$ and size $H \times W$. The z-coordinate does not have any influence on this projection. For the transformation of a BEV image from the last to the current time step, the cell centers and not the contained points are considered. Therefore, the cartesian coordinates of every cell's center are required and computed following

$$\begin{gathered} \boldsymbol{\mathcal{C}} : [1, H] \times [1, W] \subset \mathbb{N}^2 \rightarrow \mathbb{R}^3 \Rightarrow \\ \boldsymbol{\mathcal{C}}(\boldsymbol{u}) = \begin{pmatrix} \tilde{r} \cdot (u + 0.5) \cdot \cos((v + 0.5) \cdot \tilde{\alpha}) \\ \tilde{r} \cdot (u + 0.5) \cdot \sin((v + 0.5) \cdot \tilde{\alpha}) \\ 0 \end{pmatrix} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \boldsymbol{p}. \end{gathered} \tag{3.2}$$

The cartesian cell centers are then transformed from the last sensor pose $\boldsymbol{T}_{t-1}$ to the current sensor pose $\boldsymbol{T}_t$:

$$\boldsymbol{\mathcal{T}} : \mathbb{R}^4 \rightarrow \mathbb{R}^4 \;\Rightarrow\; \boldsymbol{\mathcal{T}}(\boldsymbol{p}_{t-1}) = \boldsymbol{T}_t^{-1} \cdot \boldsymbol{T}_{t-1} \cdot \begin{pmatrix} x_{t-1} \\ y_{t-1} \\ z_{t-1} \\ 1 \end{pmatrix} = \begin{pmatrix} {}^t x_{t-1} \\ {}^t y_{t-1} \\ {}^t z_{t-1} \\ 1 \end{pmatrix}. \tag{3.3}$$

The combination of these steps provides the position of the cells of the last BEV in the BEV of the current time step:

$$\begin{gathered} \boldsymbol{\mathcal{A}} : [1, H] \times [1, W] \subset \mathbb{N}^2 \rightarrow [1, H] \times [1, W] \subset \mathbb{N}^2 \Rightarrow \\ \boldsymbol{\mathcal{A}}(\boldsymbol{u}_{t-1}) = (\boldsymbol{\mathcal{P}} \circ \boldsymbol{\mathcal{T}} \circ \boldsymbol{\mathcal{C}})(\boldsymbol{u}_{t-1}) = ({}^t u_{t-1}, {}^t v_{t-1})^T = {}^t \boldsymbol{u}_{t-1}. \end{gathered} \tag{3.4}$$

This temporal transformation can then be used to transform the content of the BEV image at time $t-1$ to the BEV image at time t.

Input Alignment and Fusion The temporal transformation presented in the previous section is used for the first time at the input stage. The input memory $\boldsymbol{I}_{t-1}$ containing the aggregated point clouds until frame $t-1$ is transformed to the current time step t using the indices computed by Eq. 3.4. Cells of the memory, which lie outside the current BEV after the transformation are discarded. The transformed input memory is then fused with the input BEV $\boldsymbol{B}_t$, containing the current point cloud, see Fig. 3.1. The fusion is done by channel-wise maximum, following the PointNet encoding, which performs a channel-wise maximum over the feature vectors of all points lying inside one cell.

Feature Alignment and Fusion Following the same temporal transformation, the temporal memory $\boldsymbol{H}_{t-1}$ at feature level, which contains the aggregate output features of the past frames up to $t-1$, is transformed to the current frame. The features $\boldsymbol{F}_t$ of the current input, computed by the backbone, are then used to update the transformed temporal memory, following the residual update strategy presented in [10].

4 Experiments

4.1 SemanticKITTI

The experiments for the presented approach are conducted on the large-scale and challenging SemanticKITTI dataset [3, 11]. The single scan benchmark provides point-wise annotations of 19 classes for $360°$-Velodyne-HDL-64E scans. Over $43{,}000$ scans are divided into 22 sequences of varying length and recorded at 10 Hz. The first half of the sequences are provided with labels for training and validation while the test split is defined by the second half, with no labels published. We follow the official recommendation and use sequence 08 for validation and report the mean Intersection-over-Union (mIoU) as evaluation metric.

4.2 Implementation Details

The approach is implemented in PyTorch and trained in mixed precision mode on four Tesla V100 GPUs using distributed data parallel training.
Cross entropy and Lovász loss [4] are optimized equally weighted by Adam for $75k$ iterations. To prevent overfitting, weight decay of 0.0005 is applied as well as extensive data augmentation. Before being projected, the point clouds are randomly flipped along x- and y-axis with a probability of 0.5, randomly rotated around the z-axis and randomly cropped to a $180°$ crop. Additionally, objects of underrepresented classes, like bicycle or motorcycle, are randomly pasted into the point clouds. These objects are extracted upfront from the training set and placed on corresponding ground classes like road or sidewalk.

Backbone	TIM	TFM	mIoU (%)
✓			58.0
✓	✓		58.7
✓		✓	64.5
✓	✓	✓	**64.7**

Table 4.1: Improvements on the validation set achieved by the temporal input memory (TIM) and the temporal feature memory (TFM), compared to the single frame backbone.

Initially, the backbone is trained on single frames with a batch size of 16 and learning rate of 0.001, which decays by $e^{-5 \cdot 10^{-5} \cdot i}$. The BEV grid has a resolution of $[480, 360, 32]$. Using the pretrained backbone, the overall architecture is trained with the same batch size and learning rate and follows the temporal training proposed by [10].

4.3 Temporal BEV Segmentation

In order to evaluate the benefits of the presented recurrent temporal approach, the improvements achieved by the individual components are investigated, with the results shown in Table 4.1. The BEV backbone, which is also considered as baseline, achieves a mIoU of 58.0%. Temporal fusion at the input stage with the presented temporal input memory (TIM) improves the results to 58.7%. The fusion on feature stage has an even greater impact and considerably improves the segmentation results to a mIoU of 64.5%. Noticeably, temporal fusion of deep feature maps computed by a CNN backbone exploits temporal information and dependencies more effective than an early fusion of the input point clouds. Nevertheless, combining both stages achieves the best results and obtains an overall improvement of $+6.7\%$ in terms of mIoU.
For a more detailed analysis, the results of the individual classes are investigated and compared to the baseline, depicted in Table 4.2. Static classes are constantly improved, like fence with $+11.2\%$, and even classes with already high values benefit from temporal information. In addition, dynamic classes are considerably

Approach	mIoU	car	bicycle	motorcycle	truck	other-vehicle	person	bicyclist	motorcyclist
Backbone	58.0	93.6	43.5	60.0	54.0	42.1	56.3	73.4	10.4
TemporalBEV	**64.7**	**95.3**	**46.5**	**76.3**	**54.3**	**66.3**	**70.4**	**76.8**	**44.2**

Approach	road	sidewalk	parking	other-ground	building	fence	vegetation	trunk	terrain	pole	traffic sign
Backbone	92.2	77.9	43.3	1.2	89.0	47.0	85.6	60.0	72.7	57.7	42.9
TemporalBEV	**93.6**	**79.0**	**43.5**	**1.4**	**91.1**	**58.2**	**86.9**	**65.2**	**74.1**	**60.2**	**46.9**

Table 4.2: Results for the individual classes on the validation set of SemanticKITTI. The temporal approach outperforms the single frame backbone for every class. Values are given as IoU (%).

improved as well, especially motorcycle, other-vehicle, person and motorcyclist. This requires a deeper investigation, because movement of dynamic objects can cause alignment errors and complicates the correct temporal association. However, only fast movement causes noticeable errors, so solely a few fast moving dynamic objects do not benefit, the majority of the dynamic objects significantly benefits from the temporal information.

5 Conclusion

In this work, we presented an efficient recurrent temporal architecture for semantic segmentation of 3D point clouds relying on BEV representation. Temporal information and dependencies are exploited twice, at the input as well as feature stage. Point clouds of the last frames are aggregated to improve the information provided to the backbone. The feature maps of the backbone are then used to update a temporal feature memory, which contains the aggregated and fused features from the current frame and the past. Based on these enhanced features an improved semantic segmentation is predicted. The evaluation showed a considerable improvement achieved by the usage of temporal information and

as a result the presented approach outperforms the single frame baseline by a large margin and also for every individual class.

References

[1] Iro Armeni et al. “3D Semantic Parsing of Large-Scale Indoor Spaces”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[2] Nicolas Ballas et al. “Delving Deeper into Convolutional Networks for Learning Video Representations”. In: *arXiv* (Nov. 2016). arXiv: 1511.06432v4.

[3] Jens Behley et al. “SemanticKITTI: A Dataset for Semantic Scene Understanding of LiDAR Sequences”. In: *IEEE International Conference on Computer Vision (ICCV)*. 2019.

[4] Maxim Berman, Amal Rannen Triki, and Matthew B. Blaschko. “The Lovász-Softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks”. In: May 2018, pp. 4413–4421. eprint: 1705.08790v2.

[5] Holger Caesar et al. “nuScenes: A Multimodal Dataset for Autonomous Driving”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020.

[6] X. Chen et al. “Moving Object Segmentation in 3D LiDAR Data: A Learning-based Approach Exploiting Sequential Data”. In: *IEEE Robotics and Automation Letters(RA-L)* (2021).

[7] Ran Cheng et al. “(AF)2-S3Net: Attentive Feature Fusion With Adaptive Feature Selection for Sparse Semantic Segmentation Network”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2021, pp. 12547–12556.

[8] Christopher Choy, JunYoung Gwak, and Silvio Savarese. “4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Apr. 2019, pp. 3075–3084.

[9] Tiago Cortinhal, George Tzelepis, and Eren Erdal Aksoy. "SalsaNext: Fast, Uncertainty-aware Semantic Segmentation of LiDAR Point Clouds for Autonomous Driving". In: *International Symposium on Visual Computing (ISVC)* (Mar. 2020), pp. 207–222.

[10] Fabian Duerr et al. "LiDAR-based Recurrent 3D Semantic Segmentation with Temporal Memory Alignment". In: *International Conference on 3D Vision (3DV)* (2020), pp. 781–790.

[11] Andreas Geiger, Philip Lenz, and Raquel Urtasun. "Are we ready for autonomous driving? The KITTI vision benchmark suite". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2012.

[12] Benjamin Graham, Martin Engelcke, and Laurens van der Maaten. "3D Semantic Segmentation with Submanifold Sparse Convolutional Networks". In: 2018, pp. 9224–9232.

[13] Kaiming He et al. "Deep Residual Learning for Image Recognition". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.

[14] Qingyong Hu et al. "RandLA-Net: Efficient Semantic Segmentation of Large-Scale Point Clouds". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019.

[15] Rui Huang et al. "An LSTM Approach to Temporal 3D Object Detection in LiDAR Point Clouds". In: *IEEE European Conference on Computer Vision (ECCV)*. 2020.

[16] Yangyan Li et al. "PointCNN: Convolution On $\mathcal{X}$-Transformed Points". In: *Advances in Neural Information Processing Systems*. 2018.

[17] Wenjie Luo, Binh Yang, and Raquel Urtasun. "Fast and Furious: Real Time End-to-End 3D Detection, Tracking and Motion Forecasting with a Single Convolutional Net". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2018), pp. 3569–3577.

[18] Scott McCrae and Avideh Zakhor. "3d Object Detection For Autonomous Driving Using Temporal Lidar Data". In: *IEEE International Conference on Image Processing (ICIP)*. IEEE. 2020, pp. 2661–2665.

[19] Gregory P. Meyer et al. “LaserNet: An Efficient Probabilistic 3D Object Detector for Autonomous Driving”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019.

[20] A. Milioto et al. “RangeNet++: Fast and Accurate LiDAR Semantic Segmentation”. In: *IEEE International Conference on Intelligent Robots and Systems (IROS)*. 2019.

[21] Charles Ruizhongtai Qi et al. “PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[22] Charles Ruizhongtai Qi et al. “PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space”. In: *Advances in Neural Information Processing Systems*. 2017.

[23] Gernot Riegler, Ali Osman Ulusoy, and Andreas Geiger. “OctNet: Learning Deep 3D Representations at High Resolutions”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[24] Ahmad El Sallab et al. “YOLO4D: A Spatio-temporal Approach for Real-time Multi-object Detection and Classification from LiDAR Point Clouds”. In: *NIPS Workshop MLITS*. 2018.

[25] Hanyu Shi et al. “SpSequenceNet: Semantic Segmentation Network on 4D Point Clouds”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR*. 2020.

[26] Lyne P. Tchapmi et al. “SEGCloud: Semantic Segmentation of 3D Point Clouds”. In: *International Conference on 3D Vision (3DV)*. 2017.

[27] Hugues Thomas et al. “KPConv: Flexible and Deformable Convolution for Point Clouds”. In: *IEEE International Conference on Computer Vision (ICCV)*. 2019.

[28] Shenlong Wang et al. “Deep Parametric Continuous Convolutional Neural Networks”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2018.

[29] Jun Xie et al. “Semantic Instance Annotation of Street Scenes by 3D to 2D Label Transfer”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016.

[30] Chenfeng Xu et al. "SqueezeSegV3: Spatially-Adaptive Convolution for Efficient Point-Cloud Segmentation". In: *European Conference on Computer Vision (ECCV)*. 2020.

[31] Junbo Yin et al. "LiDAR-based Online 3D Video Object Detection with Graph-based Message Passing and Spatiotemporal Transformer Attention". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Apr. 2020, pp. 11492–11501.

[32] Fisher Yu et al. "Deep Layer Aggregation". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017.

[33] Chris Zhang, Wenjie Luo, and Raquel Urtasun. "Efficient Convolutions for Real-Time Semantic Segmentation of 3D Point Clouds". In: *International Conference on 3D Vision (3DV)*. 2018.

[34] Yang Zhang et al. "PolarNet: An Improved Grid Representation for Online LiDAR Point Clouds Semantic Segmentation". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020.

A Review on Deep Learning Approaches for Spectral Imaging

Benedikt Fischer

Vision and Fusion Laboratory
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
benedikt.fischer@kit.edu

Abstract

Deep learning algorithms have revolutionized the computer vision field in the last decade. They can reduce tedious feature engineering and have opened new possibilities of automated visual inspection. With deep learning techniques, the availability of large amounts of qualitative labeled data became more important than ever. The main share of computer vision research focuses on RGB images. With the advances in sensor technologies multi- and hyperspectral cameras have become more cost effective and accessible in recent years, allowing this imaging technology to be applied to new fields of application.
This article gives an overview of approaches to apply deep learning techniques to multi- or hyperspectral data. Several state-of-the-art methods will be reviewed and problems and difficulties will be discussed. An overview of a selection of available datasets is presented. To give a broad and diverse insight, research from different fields of application are considered, namely the remote sensing domain, the agricultural domain and the food industry.

1 Introduction

Deep learning models have shown state-of-the-art performance in computer vision problems with RGB images. They are successfully applied for classification, object detection, semantic segmentation, anomaly detection problems and more. In computer vision, most Convolutional Neural Networks (CNNs) can be broken down into two parts. The first part is called backbone and usually consists of a series of convolutional layers and pooling layers that compress the input image data into high-level feature maps. These layers act as feature extractors, meaning that certain neurons in these layers will be activated if certain features are present in the input image. While the first layers can extract basic features like edges, deeper layers can extract increasingly complex features like faces or the shape of a specific object [23]. The features extracted by the backbone can then be used as input for the second part of the neural network, which is often called head. This head depends on the task and can, for example, learn to classify the image or its individual pixels or it can output bounding boxes that correspond to the position of a specific object. To solve the problem of sample inefficiency of deep neural networks, transfer learning approaches can be used. In transfer learning, a backbone pretrained on a large dataset is used as feature extractor and finetuned on a different dataset.

This paper gives an overview of state-of-the-art approaches from the literature that can be used to apply deep learning models to multi- and hyperspectral data. A selection of research papers from different fields of application will be discussed. Different fields of application are considered in this review to recognize similarities and differences between these different domains.

First, the following section gives an overview on the principles of spectral imaging. An overview over available multi- and hyperspectral datasets from different fields of application is given in Section 3. Section 4 discusses common approaches from the literature that can be used to apply deep learning methods to spectral imaging data. The article concludes with a recap and an outlook on promising future research directions.

2 Principles of spectral imaging

Spectral imaging is the general term for multi- and hyperspectral imaging. Spectral imaging is a non-destructive measurement technique that can record images with a different spectral wavelength range and/or spectral resolution than RGB cameras. Spectral images often go beyond the visible wavelength range into the near infrared (NIR) or even the short-wave infrared (SWIR) regime. These wavelength regions are especially interesting as they allow obtaining information about the chemical composition of a material that cannot be observed by RGB cameras or the human eye. Spectral images can be represented as 3D tensors of shape (W, H, C), with the spatial width W and height H of the image and the number of spectral channels or bands C. This tensor is often called a hypercube. The values in this hypercube represent the light intensities detected by the sensor, which usually corresponds to the light reflected by a scene. However, a hyperspectral image can also be obtained for a transmission setting.

Spectral imaging is becoming more and more popular. This can also partially be attributed to the technological improvements of the sensors. Multi- and hyperspectral cameras have become much smaller and more cost effective in recent years, which allows new possible applications. This trend will most likely continue and further increase the popularity and accessibility of spectral imaging.

3 Datasets

This section lists a selection of multi- and hyperspectral datasets that are freely accessible and have been used by various researchers to compare their models. The existing multi- and hyperspectral datasets are much more diverse than RGB datasets. They have different spatial and spectral resolution and cover different spectral ranges. This makes it more complicated to compare different datasets with each other or to train one model with different datasets. The main benefit of spectral images over standard RGB images is the possibility to detect properties that are invisible to the human eye and thus cannot be detected by using RGB images. However, this also makes the labeling much more complicated. This,

and the fact that spectral cameras are much more expensive than RGB cameras contribute to the fact that the number of existing spectral datasets is much smaller than RGB datasets and also that the spectral datasets usually have fewer labeled samples. Spectral imaging is applied in many different domains like remote sensing, agricultural and food industry, medical technology [13] and recycling industry [19]. The objectives in the recycling domain are to distinguish different materials to be able to sort them. In the medical domain, disease diagnosis and image-guided surgery are relevant use cases. For example, the detection of cancer in tissues is an important topic where spectral imaging can provide added value. The remote sensing field and the agricultural and food domain are described in more detail below.

3.1 Remote Sensing

In remote sensing, hyperspectral images of the earth surface are acquired through satellites or aircrafts. These datasets are used in agriculture, environmental monitoring, urban planning and defense. Table 3.1 shows a selection of popular multi- and hyperspectral datasets that are publicly available and have been widely used by researchers in the remote sensing field. The Indian Pines (IP) [2], University of Pavia (UP), Salinas Valley (SV) and Kennedy Space Center (KSC) datasets are hyperspectral datasets with pixelwise annotations. They consist of a single image and the number of labels in table 3.1 refers to the number of pixels within that image for which a ground truth class is given. All remaining pixels are considered as background. The IP dataset was captured by the AVIRIS sensor [6] in Indiana in 1992. The KSC dataset and the SV dataset were also captured by the AVIRIS sensor in Florida and in California respectively. The UP dataset was recorded by the ROSIS sensor [10] over the campus of the University of Pavia and has the highest spatial resolution with 1.3 m per pixel. The IP and UP datasets are available online on the GRSS Data and Algorithm Standard Evaluation (DASE) website (http://dase.grss-ieee.org). More remote sensing dataset are summarized in the works of [1] and [15].

EuroSAT [9] is a multispectral dataset, created with the freely accessible Sentinel-2 satellite images. The dataset is patch-based, it consists of 27000 small image patches that contain one predominant class and thus have one ground truth class

Table 3.1: Details of five popular remote sensing datasets. The Indian Pines (IP), University of Pavia (UP), Salinas Valley (SV) and Kennedy Space Center (KSC) are hyperspectral datasets with pixelwise classification labels and the EuroSAT dataset is a multispectral dataset with imagewise classification labels.

Dataset Name	IP	UP	SV	KSC	EuroSAT
Pixels	145 ×145	610 ×340	512 ×217	512 ×614	64 ×64
Bands	224	103	227	176	13
Spectral Range in nm	400-2500	430-860	400-2500	400-2500	440-2200
Spatial Resolution in m	20	1.3	3.7	18	10
Classes	16	9	16	13	10
Labels	10,249	42,776	54,129	5,202	27,000

label each. Thus, this dataset does not allow evaluating per-pixel segmentation algorithms; however, this is not its intention. Since the Sentinel-2 satellite is scanning earth's surface repeatedly approx. every five days, a classifier trained on this data could be used to continuously monitor land surfaces and detect changes in land use.

3.2 Agriculture and food industry

The most common objectives of spectral data in the agricultural domain are monitoring the state of plants, crops, fruits and vegetables. Examples are the detection of diseases and weeds or the estimation of ripeness. In the food domain, common objectives are the detection of defects like bruises and the prediction of physical parameters like acid and sugar content in fruits and vegetables. In the food domain spectrometry has a long history, thus many spectral datasets exist of point measurements without spatial dimensions. Multi- and hyperspectral datasets have become more popular in recent years, but still many dataset of the food domain are not made publicly available. It is also noteworthy that there seem to be no popular benchmark datasets that are widely used by researchers like they exists in the remote sensing domain.

Varga et al. [21] recently published a dataset that contains hyperspectral images of avocados and kiwis and covers different ripening states from unripe to overripe.

The fruits were recorded with two cameras simultaneously: the Specim FX 10 with 224 channels from 400 to 1000 nm and an INNO-SPEC Redeye 1.7 with 252 channels from 950 to 1700 nm. The images were cropped to contain one fruit and have varying spatial dimensions of around 200 to 300 pixels in each dimension. In total the dataset contains 1038 recordings of avocados and 1522 recordings of kiwis. A subset of 180 avocado images and 262 kiwi images were annotated with the reference labels: weight, weight loss during storage, storage time, firmness determined with a penetrometer and ripeness level determined by appearance and taste.

The Ladybird Brassica dataset [3] contains image data, based on weekly scans of cauliflower and broccoli vegetables over a 10-week period from transplant to harvest. This multimodal dataset consists of stereo vision data, thermal images and hyperspectral images. The hyperspectral images were recorded with the Resonon Pika XC2 camera, with 447 channels in the range 400-1000 nm. The crops were annotated with bounding boxes.

4 Deep Learning Methods

Deep learning refers to models that use neural network with many layers. Deep learning methods and more specifically deep CNNs have shown state-of-the-art performance in computer vision problems like classification, object detection, semantic segmentation and anomaly detection. The convolutional layers consist of many filters that act as spatial-spectral feature extractors. In the early layers simple features like edges can be learned by the network, whereas deeper layers can extract more complex features like specific textures or geometries. Convolutional layers are so efficient for image data due to their translation invariance in the spatial dimension. A filter that learned to extract a specific feature will extract this feature regardless of its spatial location in the image. This idea is also called weight sharing and reduces the required weights dramatically compared to a fully connected layer. The convolutional filter exploits the local correlation in the spatial dimension of the image. Convolutional networks have been show to work well with grayscale images and 3-channel RGB images, but when working with multi-channel images the size of the filters in the first layer

increases drastically if normal 2D-CNN filters are used. Just like adjacent pixels, adjacent spectral bands are also correlated. A 2D-CNN filter does not make full use of this spectral correlation. One possible solution to this problem are 3D-CNN filters, however they cannot detect long-range dependencies in the spectral dimension sufficiently. A possible solution to this problem might be the use of attention-based methods.

Another problem of all these models is sample efficiency. As discussed in Section 3, the available spectral datasets have fewer samples than popular RGB datasets like ImageNet. In computer vision with RGB images, transfer learning has shown to be a powerful tool to improve sample efficiency. Models that have been trained on datasets with millions of images can be finetuned on much smaller datasets and still achieve good results. This section presents some methods that try to make use of transfer learning to apply RGB pretrained models to spectral imaging data. Another promising method to improve sample efficiency are unsupervised learning approaches.

The selection of a suitable and efficient model for spectral images poses a challenge. This section shows a selection of different approaches to these problems from the literature and discusses their results.

4.1 Preprocessing

Unlike traditional machine learning models, neural networks are usually more robust with respect to data preprocessing. Thus, most works do not use preprocessing for the spectral imaging data, with the exception of normalization. Common image normalization techniques are to normalize all values to the range $[0, 1]$ or to normalize the first- and second-order moments to obtain a zero mean and unit variance [1]. This normalization can be done for each channel independently or for all channels globally.

4.2 2D CNN

To make standard 2D-CNN architectures work with spectral imaging data, that has more than 3 channels, either the number of channels of the input hypercube

has to be reduced before it is fed to the network or the first layer of the network has to be modified. To reduce the number of channels, different methods can be used: representative wavelengths can be selected with feature selection algorithms, the spectral dimension can be compressed with dimensionality reduction techniques like Principal Component Analysis (PCA) or a compression layer with learnable weights can be added before the first layer.

4.2.1 Wavelength selection

Often the most useful wavelengths for the task at hand get selected with feature selection algorithms and then only those selected wavelengths are used as input for a model. This approach solves the problem of the high dimensional data by reducing it to a few wavelengths. However, this approach usually requires tedious feature engineering and does not generalize well as these wavelengths are chosen to work optimal for one specific problem. Gao et al. [5] recorded hyperspectral images of 120 strawberries and classified them into ripe and early ripe. They use a sequential feature selection algorithm to select a feature wavelength and input this as a grayscale image into an AlexNet Model.
Pang et al. [14] recorded 300 hyperspectral images of bruised apples with 256 wavelengths from 930 to 2548 nm. They use wavelength selection to compress the data to 3 channels and then apply a YOLOv3 object detection model. To extract the effective wavelengths they applied PCA to broad regions of the spectrum and visually selected the principle component (PC)-image with the most apparent contrast between sound and bruised tissue. Then they chose 3 wavelengths where the weighing coefficient curve of the PC-image had extreme values. They also compared the result with a traditional segmentation algorithm and found the deep learning approach with YOLOv3 is more robust.

A common option to reduce the number channels in spectral images is the use of PCA. The grayscale PC-images can be concatenated to one image. However, some research has found that this approach does not perform very well when used as input for CNNs. Varga et al. [21] tested different architectures with different input data: a full hyperspectral image, a pseudo-RGB image and PC-images of the full spectrum. They found that the PC-images do not perform as well as RGB images, even when using non-pretrained CNN architectures. They conclude

that PCA might remove some necessary information that is still available in the pseudo-RGB images.

Zhao and Du [25] implemented a hybrid approach. They use PCA to reduce the spatial dimension of the UP dataset and feed this to a 2D-CNN to extract deep spatial features. They also implement a balanced local discriminant embedding (BLDE) in parallel to extract spectral features from the hyperspectral data. Finally, they stack both spectral and spatial features together and use a LR classifier to classify the pixels of the UP dataset with an accuracy of 96%.

4.2.2 Added trainable layers

Steinbrener et al. [20] added two 2D-CNN layers in front of a pretrained GoogLeNet network to reduce the number of channels from 16 to 3. They use a custom dataset with 2700 multispectral images of 13 different classes of fruits and vegetables with 16 spectral bands for their finetuning. This method shows better results for their dataset than using the pretrained GoogLeNet with pseudo-RGB images, which shows the added value of additional wavelengths. However, they do not compare the results with a non-pretrained GoogLeNet, thus the benefit of transfer learning cannot be evaluated with their paper.

Zhang et al. [24] utilized a VGG16 backbone, pretrained on ImageNet, to segment bruises in blueberries from hyperspectral transmittance images. Their dataset consists of 1200 hyperspectral images with pixelwise labels for the 4 classes bruised, unbruised, calyx and background. To use the hyperspectral images with a pretrained backbone, they added a convolutional layer before the first layer to reduce the number of channels from 87 to 3. They found that using the full spectrum with 87 channels achieved better accuracy than using only 3 or 9 selected wavelengths. They also compared the results of the pretrained backbone with a backbone that was trained from scratch with randomly initialized weights. For their dataset, the model that was trained from scratch performed better than the pretrained network. The reason for this might be that the output of the added first layer has a different distribution than the original input images of the pretrained model. Since the learned filters in deeper layers depend on the output feature maps of previous layers, those learned filters might be less useful if the input distribution changes. They conclude that a

possible solution for this problem could be to train only the added layer and freeze the other layers to maximize similarity between the distribution of the added layer outputs and the original input images. Another possible reason for the poorer performance of the pretrained model could be that their use case of segmenting bruised regions within blueberries might differ too much from the objective of the pretrainig, which for ImageNet is classifying images based on more than 20000 categories. The blueberry bruises in this dataset apparently do not have distinct spatial patterns, unlike most categories in ImageNet. Thus, the model might benefit less from the pretraining. It would be interesting to test if a backbone pretrained on a segmentation dataset instead of a classification dataset would achieve better results.

4.2.3 Modified first layer

Wang et al. [22] use a modified ResNet architecture where the first 2D-CNN layer has been modified to work with input images that have 151 channels instead of 3. Their custom dataset contains 557 hyperspectral transmittance images of blueberries, which are classified into good and bruised. They resize the hyperspectral cubes from $(128, 128, 1002)$ to $(32, 32, 151)$ to reduce the computational complexity and feed this reduced hypercube to their model. They achieve an accuracy of of 88% in classifying the bruised samples, which was better than the result of traditional machine learning models like linear regression or random forest. However, they highlight the limitations of traditional 2D convolutional layers when working with multi-channel images. A 2D convolutional filter uses every channel of the input data, which does not fully exploit the local correlation between channels and introduces many unnecessary weights to be trained. This may lead to overfitting and harm the generalizing ability of the model. This is especially the case for small datasets, which are common for hyperspectral data.

4.3 2D + 1D CNN

The combination of 2D and 1D CNNs is also called depthwise separable convolution and can reduce the computational complexity and the number

of weights, compared to pure 2D or 3D CNNs. The depthwise separable convolution consists of a depthwise (DW) convolution followed by a pointwise (PW) convolution (or PW followed by DW) [7]. The depthwise convolution allows to model spatial relationship by applying a 2D filter-kernel to each input channel. This allows learning different spatial features for different channels. The pointwise convolution is a 1×1 convolution that can model relationships across channels. A 1×1 convolution can also be used to reduce the number of channels of an input tensor, which will result in a reduction of the number of parameters in the following convolution layer. They can also be referred to as squeeze layers [18].

Depthwise separable convolution layers have been applied by Varga et al. [21] to classify the fruit ripening dataset that was described in section 3.2. They compared their model with a ResNet and an AlexNet architecture whose first 2D-CNN layers have been adapted to the size of the hyperspectral input data and found that the separable convolution outperformed the 2D-CNNs.

Senecal et al. [18] propose a SpectrumNet architecture to classify the EuroSat dataset. They also compared the use of depthwise separable convolutions with standard 2D-CNN. While the final classification accuracy of both CNNs was similar, the standard 2D-CNN was more sample efficient in their case. The decoupling of cross-channel correlations and spatial correlation seems to make the training more difficult. However, the use of the depthwise separable convolution reduced the computational requirements of the network significantly which could be a worthwhile trade-off.

4.4 3D CNN

While a 2D convolutional filter is sliding across the two spatial dimensions and produces a 2D feature map, a 3D convolutional kernel is sliding across all three dimensions of the hypercube and produces 3D feature cubes. Like with 2D-CNNs a layer can contain several filter kernels, in which case several feature cubes are created as output of that layer. Li et al. [11] and Chen et al. [4] achieved competitive results to state-of-the-art models on the IP and UP datasets, using 3D CNNs.

To reduce the computational complexity of standard 3D-CNNs, [17] proposes a hybrid spectral CNN (HybridSN) for HSI classification that achieves state-of-the-art performance on the IP dataset. The HybridSN is a 3D-CNN followed by 2D-CNN. The 3D-CNN can extract joint spatial-spectral features from the input image and the following 2D-CNN can learn more abstract spatial features.

4.5 Attention

One disadvantage of CNNs is that they are not good at modelling long-range dependencies. However, the spectra of hyperspectral images do contain long-range dependencies as the wavelengths are correlated and may contain hundreds of channels. To solve this problem, different attention-based models have been proposed recently [8], [16].

An Attention-Based Adaptive Spectral-Spatial Kernel (A2S2K) ResNet has been proposed by [16] very recently and has achieved state-of-the-art performance on the KSC dataset with an overall accuracy of 99.43% and competitive results to the state-of-the-art on the UP and IP datasets.

Zhu et al. [26] recently proposed a spectral-spatial dependent global learning (SSDGL) model that uses an attention mechanism as well as a convolution long short-term memory module. Their model achieved state-of-the-art performance on the UP dataset and competitive results to the state-of-the-art on the IP dataset.

4.6 Unsupervised Methods

Unsupervised learning is a promising approach to the problem of limited availability of labeled data in spectral imaging. Unsupervised methods can make use of unlabeled data and the amount of available unlabeled data is much higher than the amount of labeled data. A common unsupervised method are autoencoders. Autoencoders are composed of an encoder that compresses the input data into latent feature space and a decoder that gets the latent space as input and tries to reconstruct the original input data from it. The encoder can use different convolutional layers and pooling layers to compress the data. If the autoencoder is trained with the unlabeled data, the encoder can be used as a

feature extractor that will ideally be able to extract meaningful spatial-spectral features. Such an autoencoder can be used to extract features, e.g. from a much smaller labeled dataset, which can then be used as input for a supervised classifier. This approach is called semi supervised learning. Liu et al. [12] use a similar approach to classify the UP dataset and achieves competitive results to state-of-the-art methods.

4.7 Perspectives

Many different deep learning architectures for spectral imaging have been published in recent years and have set a new state-of-the-art performance in the field. Especially in the remote sensing domain, a lot of research has been published. This may also be partially contributed to the availability of several widely used benchmark datasets. Such benchmark datasets allow researchers to compare their models in a competitive way. However, the current state-of-the-art models are reaching accuracies close to 100% on some of those datasets. This may indicate the need for new and potentially more diverse or more difficult datasets. To the knowledge of the author, no such widely used benchmark datasets exist in the agricultural or food domain. In fact, many researchers just use their own private datasets. These domains could benefit from more public benchmark datasets.

Apart from the data, there is also a lot of potential for improvement on the model side. For example, the sample efficiency and robustness of such models offers room for improvements. The existing hyperspectral datasets are very different in spatial size and resolution and in the spectral wavelength range and resolution. It could be beneficial to have a universal model that works for datasets with different resolution without the need to modify its architecture, similar to RGB models that work for different spatial resolutions. Such a model could be trained with a combination of multiple different datasets, which would massively increase the available data. Unsupervised models also have a great potential, since they do not need labeled data.

5 Conclusion

Deep learning models have proven to be efficient for multi- and hyperspectral data. Many different convolutional architectures have been proposed to process spectral imaging data. The 2D, 2D + 1D and 3D CNNs combine spatial and spectral information in an intuitive and efficient way. They show state-of-the-art performance on different multi- and hyperspectral datasets. CNNs that use an attention mechanism to be able to model long-range dependencies are becoming more popular lately, also showing state-of-the-art performance on the available datasets. One of the main remaining challenges is the scarce availability of large annotated datasets. More and bigger spectral imaging datasets, similar to the ImageNet and COCO datasets used in RGB imaging, would be beneficial for the research field. However, the labeling of such data is very time consuming. Thus, a promising direction is the development of unsupervised and semi-supervised approaches.

References

[1] Nicolas Audebert, Bertrand Saux, and Sébastien Lefèvre. "Deep Learning for Classification of Hyperspectral Data: A Comparative Review". In: *IEEE Geoscience and Remote Sensing Magazine* 7 (Apr. 2019). DOI: 10.1109/MGRS.2019.2912563.

[2] Marion F. Baumgardner, Larry L. Biehl, and David A. Landgrebe. *220 Band AVIRIS Hyperspectral Image Data Set: June 12, 1992 Indian Pine Test Site 3*. Sept. 2015. DOI: doi:/10.4231/R7RX991C.

[3] Asher Bender, Brett Whelan, and Salah Sukkarieh. "A High-Resolution, Multimodal Data Set for Agricultural Robotics: A Ladybird's-Eye View of Brassica". In: *Journal of Field Robotics* 37.1 (2020), pp. 73–96. ISSN: 1556-4967. DOI: 10.1002/rob.21877.

[4] Yushi Chen et al. “Deep Feature Extraction and Classification of Hyperspectral Images Based on Convolutional Neural Networks”. In: *IEEE Transactions on Geoscience and Remote Sensing* 54.10 (Oct. 2016), pp. 6232–6251. ISSN: 1558-0644. DOI: 10.1109/TGRS.2016.2584107.

[5] Zongmei Gao et al. “Real-Time Hyperspectral Imaging for the in-Field Estimation of Strawberry Ripeness with Deep Learning”. In: *Artificial Intelligence in Agriculture* 4 (Jan. 2020), pp. 31–38. ISSN: 2589-7217. DOI: 10.1016/j.aiia.2020.04.003.

[6] Robert O Green et al. “Imaging Spectroscopy and the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS)”. In: *Remote Sensing of Environment* 65.3 (Sept. 1998), pp. 227–248. ISSN: 0034-4257. DOI: 10.1016/S0034-4257(98)00064-9.

[7] Jianbo Guo et al. “Network Decoupling: From Regular to Depthwise Separable Convolutions”. In: *arXiv:1808.05517 [cs]* (Aug. 2018). arXiv: 1808.05517 [cs].

[8] Xin He, Yushi Chen, and Zhouhan Lin. “Spatial-Spectral Transformer for Hyperspectral Image Classification”. In: *Remote Sensing* 13.3 (Jan. 2021), p. 498. DOI: 10.3390/rs13030498.

[9] Patrick Helber et al. “EuroSAT: A Novel Dataset and Deep Learning Benchmark for Land Use and Land Cover Classification”. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 12.7 (July 2019), pp. 2217–2226. ISSN: 2151-1535. DOI: 10.1109/JSTARS.2019.2918242.

[10] B. Kunkel et al. “ROSIS (Reflective Optics System Imaging Spectrometer) - A Candidate Instrument For Polar Platform Missions”. In: *1987 Symposium on the Technologies for Optoelectronics*. Ed. by C. Stuart Bowyer and John S. Seeley. Cannes, France, Apr. 1988, p. 134. DOI: 10.1117/12.943611.

[11] Ying Li, Haokui Zhang, and Qiang Shen. “SpectralSpatial Classification of Hyperspectral Imagery with 3D Convolutional Neural Network”. In: *Remote Sensing* 9.1 (Jan. 2017), p. 67. DOI: 10.3390/rs9010067.

[12] Bing Liu et al. “A Semi-Supervised Convolutional Neural Network for Hyperspectral Image Classification”. In: *Remote Sensing Letters* 8.9 (Sept. 2017), pp. 839–848. ISSN: 2150-704X. DOI: 10.1080/2150704X.2017.1331053.

[13] Guolan Lu and Baowei Fei. “Medical Hyperspectral Imaging: A Review”. In: *Journal of Biomedical Optics* 19.1 (Jan. 2014), p. 010901. ISSN: 1083-3668, 1560-2281. DOI: 10.1117/1.JBO.19.1.010901.

[14] Qi Pang et al. “Detection of Early Bruises on Apples Using Hyperspectral Imaging Combining with YOLOv3 Deep Learning Algorithm”. In: *Journal of Food Process Engineering* n/a.n/a (2021), e13952. ISSN: 1745-4530. DOI: 10.1111/jfpe.13952.

[15] M. E. Paoletti et al. “Deep Learning Classifiers for Hyperspectral Imaging: A Review”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 158 (Dec. 2019), pp. 279–317. ISSN: 0924-2716. DOI: 10.1016/j.isprsjprs.2019.09.006.

[16] Swalpa Kumar Roy et al. “Attention-Based Adaptive SpectralSpatial Kernel ResNet for Hyperspectral Image Classification”. In: *IEEE Transactions on Geoscience and Remote Sensing* 59.9 (Sept. 2021), pp. 7831–7843. ISSN: 1558-0644. DOI: 10.1109/TGRS.2020.3043267.

[17] Swalpa Kumar Roy et al. “HybridSN: Exploring 3-D2-D CNN Feature Hierarchy for Hyperspectral Image Classification”. In: *IEEE Geoscience and Remote Sensing Letters* 17.2 (Feb. 2020), pp. 277–281. ISSN: 1558-0571. DOI: 10.1109/LGRS.2019.2918719.

[18] Jacob J. Senecal, John W. Sheppard, and Joseph A. Shaw. “Efficient Convolutional Neural Networks for Multi-Spectral Image Classification”. In: *2019 International Joint Conference on Neural Networks (IJCNN)*. Budapest, Hungary: IEEE, July 2019, pp. 1–8. ISBN: 978-1-72811-985-4. DOI: 10.1109/IJCNN.2019.8851840.

[19] Silvia Serranti, Roberta Palmieri, and Giuseppe Bonifazi. “Hyperspectral Imaging Applied to Demolition Waste Recycling: Innovative Approach for Product Quality Control”. In: *Journal of Electronic Imaging* 24.4 (July 2015), p. 043003. ISSN: 1017-9909, 1560-229X. DOI: 10.1117/1.JEI.24.4.043003.

[20] Jan Steinbrener, Konstantin Posch, and Raimund Leitner. “Hyperspectral Fruit and Vegetable Classification Using Convolutional Neural Networks”. In: *Computers and electronics in agriculture* (2019). ISSN: 0168-1699.

[21] Leon Amadeus Varga, Jan Makowski, and Andreas Zell. “Measuring the Ripeness of Fruit with Hyperspectral Imaging and Deep Learning”. In: *2021 International Joint Conference on Neural Networks (IJCNN)*. July 2021, pp. 1–8. DOI: 10.1109/IJCNN52387.2021.9533728.

[22] Zhaodi Wang, Meng-Han Hu, and Guangtao Zhai. “Application of Deep Learning Architectures for Accurate and Rapid Detection of Internal Mechanical Damage of Blueberry Using Hyperspectral Transmittance Data”. In: *Sensors (Basel, Switzerland)* 18 (Apr. 2018). DOI: 10.3390/s18041126.

[23] Matthew D. Zeiler and Rob Fergus. “Visualizing and Understanding Convolutional Networks”. In: *Computer Vision ECCV 2014*. Ed. by David Fleet et al. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2014, pp. 818–833. ISBN: 978-3-319-10590-1. DOI: 10.1007/978-3-319-10590-1_53.

[24] Mengyun Zhang et al. “Fully Convolutional Networks for Blueberry Bruising and Calyx Segmentation Using Hyperspectral Transmittance Imaging”. In: *Biosystems Engineering* 192 (Apr. 2020), pp. 159–175. ISSN: 15375110. DOI: 10.1016/j.biosystemseng.2020.01.018.

[25] Wenzhi Zhao and Shihong Du. “SpectralSpatial Feature Extraction for Hyperspectral Image Classification: A Dimension Reduction and Deep Learning Approach”. In: *IEEE Transactions on Geoscience and Remote Sensing* 54.8 (Aug. 2016), pp. 4544–4554. ISSN: 1558-0644. DOI: 10.1109/TGRS.2016.2543748.

[26] Qiqi Zhu et al. “A Spectral-Spatial-Dependent Global Learning Framework for Insufficient and Imbalanced Hyperspectral Image Classification”. In: *IEEE Transactions on Cybernetics* (2021), pp. 1–15. ISSN: 2168-2267, 2168-2275. DOI: 10.1109/TCYB.2021.3070577.

Dynamic Planning Pipeline for Indoor Inspection Flights

Raphael Hagmanns

Vision and Fusion Laboratory
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
raphael.hagmanns@kit.edu

Abstract

In this report a basic pipeline for planning and operating an indoor drone flight is presented and evaluated in detail. We introduce the structure and interface considerations of a Planner Manager enabling autonomous indoor flights. The interaction routines of different planners are introduced in detail before we evaluate the system in both simulation and real test flights. We show that the system is capable of managing the typical building blocks of a mobile robotics system. Most of the components can be swapped easily to allow for rapid prototyping without the need to rework the whole pipeline.

1 Introduction

UAVs (Unmanned Aerial Vehicles) have become increasingly important in the last couple of years. Both industrial and consumer sectors require UAVs to be equipped with more and more intelligent and autonomous behavior. This includes automatic obstacle avoidance, efficient path planning as well as an environment sensing with varying imaging sensors. While some of these

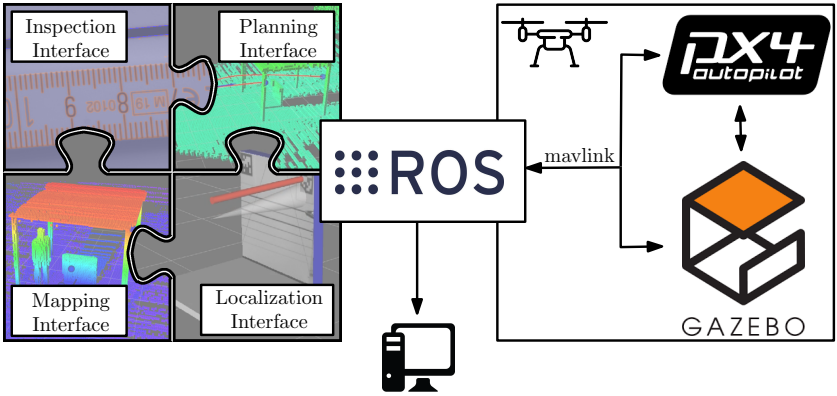


Figure 1.1: Overview of different pipeline building blocks.

functionalities are included in flight control software [1],[2], they rarely meet the requirements of the research community. Prototype driven development of new algorithms, reproducible testing in simulation as well as in practical experiments and fine graded control over all system variables often require researchers to create their own entire software stack in order to run a specific algorithm. Due to the novelty of the research area, many of the basic requirements as *feedback control*, *trajectory generation* or *state estimation* are still very active research topics [2]. Without access to the whole deployment stack, it is often difficult to run a specific module or to reproduce the results. On the other hand, if the stack is available, extensive modifications are often required in order to test it against alternative implementations.

Therefore, we propose a framework which allows for a loose coupling of several main building blocks of a mobile robot autonomy stack. These building blocks are depicted in 1.1. We target autonomous indoor inspection flights with some specific inspection target, which comes with the additional difficulty of accurate state estimation. Usually some kind of visual inertial odometry must be provided

[1] `https://www.dji.com/de/guidance`
[2] `https://docs.px4.io/v1.9.0/en/advanced_features/`

to allow a drone to fly in such GPS-denied environments. However, to support different environments, we don't want to restrict the platform to a specific source of odometry as explained in the localization interface (see Section 3.1). The generation of a obstacle free trajectory requires a planner to interact with some kind of map representation, this interaction is described in further detail in Section 3.1. We then continue to describe the main control routine in further detail in Section 3.2. Finally, we include some practical considerations in Section 4 and present an evaluation of the controller pipeline and the interfaced algorithms.

The Robot Operating System (ROS) serves as main link between the different components in our system as visualized in Figure 1.1. It is widespread in the robotic community and comes with a powerful toolset consisting of sensor drivers, simulation frameworks (Gazebo) and a selection of state-of-the-art perception algorithms. [3],[4] The node based architecture together with a publisher / subscriber information scheme allows for a semantic abstraction of single building blocks into reusable, modular software packages. Our platform is intended to be deployed on drones running the px4 software stack.[5] Px4 is an open source autopilot working on various hardware platforms. It comes with *SITL* and *HITL* features and supports standardized communication via the mavlink communication protocol. Additionally it allows to publish and retrieve odometry information to and from ros nodes via *mavlink*[6], which makes it the most popular platform in the research community.

2 Related Work

There exist few software stacks containing a high level controller which is able to plan and execute a given waypoint trajectory. Most of the available dynamic UAV planners solely rely on a specific underlying map representation and require additional work to run in a real environment.

[3] `https://ros.org/`
[4] `https://gazebosim.org/`
[5] `https://px4.io/`
[6] `https://mavlink.io/`

In [12], Schmid et al. present a generic planning framework with the focus on *active* planning. While the planning module itself is highly configurable, the underlying map representation must be Voxblox [9] and the calculation of the exploration gain of specific viewpoints is deeply interlinked with the raycasting for volumetric map building. The planner is working in the *RotorS* [4] environment, which on of the most common simulation frameworks for UAVs. However, the framework is not meant to be used for flights in practice. *Fuel* [15] is another explorative path planner based on a *frontier information structure* maintaining a tree of already explored paths. They use a hierarchical planner structure which is able to perform global and local planning. It however lacks the possibility to replace single parts of the planner and does not come with integrated controlling capabilities.

The flight controller software stack *px4* itself provides different modes suitable for simple waypoint navigation. This comes with the advantage of a strong link to the flight controlling mechanisms. Theses modes are only provided for GPS based flights and the support for local planning is limited. Within the px4-universe, the recently open-sourced frameworks XTDrone [14] and MRS UAV System [2] offer a variety of functions, platforms and sensors and are highly extensible by providing a plugin based architecture. The MRS UAV System even allows to swap between different odometry sources, trackers and controllers mid-flight making it an advanced platform for carrying out research experiments in simulation and real world scenarios. Due to the sheer size and complexity of the platforms, it is not a trivial task to adapt or exchange single components in the systems.

Our proposed architecture does not aim on providing a planner which is on pair with state-of-the-art planners but rather allow the utilization of different modules with unified interfaces and minimal overhead.

3 Pipeline

In the following, we briefly describe our pipeline architecture and discuss the planning routines in greater detail. The main building blocks of the pipeline are depicted in Figure 3.2. Figure 3.1 introduces the notation of the drone's reference

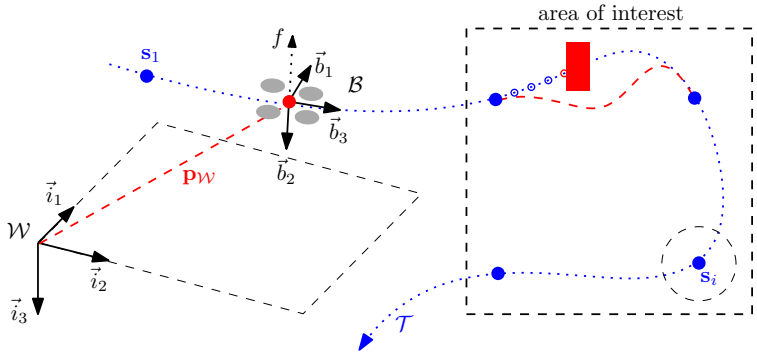


Figure 3.1: UAV System Overview. Similar to [7], the UAV is represented through its body frame $\mathcal{B} = \{\vec{b}_1, \vec{b}_2, \vec{b}_3\}$ originated in the center of mass (red dot) with the combined thrust of the four rotors pointing in $-\vec{b}_2$ direction. The UAVs position estimate is given through $\mathbf{p}_\mathcal{W}$, w.r.t. the reference world frame $\mathcal{W} = \{\vec{i}_1, \vec{i}_2, \vec{i}_3\}$. The drone is tracking the given trajectory $\mathcal{T} = \mathbf{s}_1, \ldots, \mathbf{s}_n$, while dynamically avoiding the obstacle marked in red (cf. Section 3.1). $\mathcal{T}$ is calculated to be a dynamically feasible trajectory, guaranteed to contain the blue inspection points unless they lie within an obstacle. The circle around $\mathbf{s}_i$ depicts the *acceptance radius* for the respective waypoint.

pose and frames. As input we take a list of setpoints $\mathcal{T} = \mathbf{s}_1, \ldots, \mathbf{s}_n$ given as coordinates in a specific reference system $\mathcal{W}$, where each setpoint $\mathbf{s}_i = [\mathbf{t}, \psi]^\top$ consists of a position $\mathbf{t} = (x, y, z)^\top$ and a heading ψ. This input can either be user defined or the result of some static planning routine as explained in Section 3.1. Final output of the pipeline are the lowlevel *setpoint* commands which then serve as input for the flight controller.

3.1 Architecture

The controller handles different components which now will be explained in further detail. Most of the building blocks follow a plugin architecture so that they can easily replaced with alternatives as long as they are implementing the same interface.

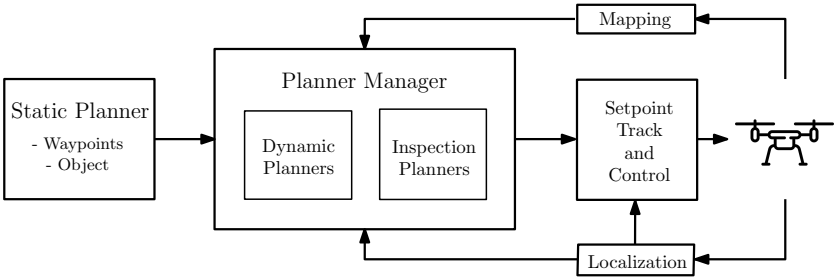


Figure 3.2: Planning Blocks for a Pipeline. An initial trajectory can be provided by a static planner or a waypoint list. The Planner Manager handles different kinds of subplanners, which all use an interface to an underlying map representation and the current position estimate. The "Setpoint Track & Control" block receives the output from the localization system and provides inputs to the low-level flight controller. For a more detailed description of the building blocks, see Section 3.1.

Static Planner For some inspection tasks, it can be useful to pre-generate an inspection trajectory with the goal of optimal inspection coverage while reducing path length and uncertainty at the targeted measuring points at the same time. To support such a pre-computation using a triangle mesh as input, we adopted a version of the Structural Inspection Planner [1] and abstracted it into a single "Static Planner" node. Output of this procedure is a set of optimized viewpoints optimizing the criteria stated above. For details we refer to the original publication in [1].

Localization Interface We don't target a specific localization system within our pipeline. We require some source of odometry to provide an estimate of the pose $\mathbf{p}_{\mathcal{W}} = (\mathbf{t}, \mathbf{R})$ of the vehicle. The typical source of such odometry is the GPS / IMU fusion provided by the px4 flight controller utilizing an Extended Kalman Filter. Other sources of position estimates such as Visual Odometry (cf. Section 4) system or a full SLAM approaches might be used as well. The position estimate is provided to the high-level planner manager as well as to the *Setpoint Control* block.

Setpoint Track and Control This building block generates the commands for the low level flight controller as output. It is setup to either use the flight controller internal position controller by providing the current target

setpoint $\mathbf{s} = [\mathbf{t}, \psi]$ at a rate of 50hz. It additionally supports the utilization of a $\mathbf{SE}(3)$ geometric controller as described in [7] to support more aggressive maneovers. We refer to that publication [7] for a detailed description of the assumed UAV dynamic model. In that mode, output will be provided as *attitude rate* commands right into the *attitude controller* of the flight controller. However, using a geometric controller requires a smooth and feasible input trajectory [7]. Following the approach by Richter, Bry, and Roy [11], we consider the construction of such a piecewise polynomial trajectory out of a set of waypoints to be a linear optimization problem and solve it using an unconstraint linear solver. The implementation in [3] provides additional methods to check for the trajectory feasibility.

Mapping Interface Typically, planners work on an underlaying map representation to account for obstacles. This representation may either be derived from an existing environment or it can be constructed dynamically using the current position estimate $\mathbf{p}_{\mathcal{W}}$ together with some sort of depth sensor information. The pipeline does not rely on a specific type of such representation as long as it supports obstacle-queries for a bounding volume $\mathbf{b} = (\mathbf{t}, \mathbf{d}) \in \mathbb{R}^{2\times 3}$ of the size $\mathbf{d}$. Most of the available volumetric map representations allow for such an operation, in particular we implemented the interface for some of the most popular representations: OctoMap [6] as efficient Octree based volumetric map; Voxblox, a *Truncated Signed Distance Field* based representation [9] and EWOK [13], a highly efficient voxel representation based on a local ringbuffer.

Constraint Planner Sometimes it is useful to constrain the UAV trajectory to a specific operation area. While this could be modeled similar to obstacles in the underlying map representation, we decided to abstract this operation into a different planner, which can reject specific points outside of the defined areas and re-trigger the trajectory generation process with additional constraints. It can also be used to provide warnings or trigger *safety actions* if a no-fly zone is approached.

Dynamic Planner With an underlaying map representation in place, a dynamic planner can now be put into use to dynamically avoid obstacles during

execution time. We leverage the *Open Motion Planning Library* (OMPL) [7] as default framework for local planning as it enables us to compare different kinds of planners using the same interface. In our default configuration, we use a *Informed* RRT* Planner [5], which has proven to outperform the classical RRT* in terms of both convergence rate and solution quality. Each state in the planners search space is being checked for potential collision using an approximate *hit bounding volume* of the vehicle. The resulting trajectory is post-processed using a path simplification and smoothing routine.

Inspection Planner As we are targeting object inspection we introduce another module, which is solely responsible for planning low-level inspection routines. This allows us, for example, to trigger a hover action when we reach a point of interest. We can also use it to perform a static maneuver in order to generate multiple views of a target inspection point. When carrying out an object measurement task, these measures help to a reduce the measurement uncertainty, which otherwise would be mainly influenced by shutter and motion blur effects. When activated during hovering the Inspection Planner may also target a specific distance to an inspection object and overrule the safety distances maintained by the Dynamic Planner. This can prove useful if an inspection unit needs a specific working distance to perform a specific task.

3.2 Planning Procedure

We now describe the interaction of the modules introduced in Section 3.1 in further detail. To ensure a high robustness of the planning system, we employ a multithreaded structure. The *Setpoint Track and Control* block is publishing commands with a high availability while the replanning procedure is running at a lower frequency and the cascading planner structure is invoked on demand. Algorithm 3.1 explains the concept of the *horizon*-based dynamic planning procedure. Multiple planners may be used in a *Planner Manager* $\mathcal{P}$, their

[7] https://ompl.kavrakilab.org/

configurations or even the whole planner may be replaced at runtime. For each update, the previously generated smooth trajectory $\mathcal{T}$ is being evaluated for feature waypoints up to the horizon h (line 7) and stored in a planning queue Q (lines 1-3). If the evaluated position lies within the *acceptance radius* of the currently targeted inspection point, we mark it as reached (line 8) and set a new goal. All registered planners now *check* the generated points (lines 9-13). The *check* depends on the nature of the planner, it typically is an obstacle query against the *Mapping Interface*. Within the check, planners can change the *state* of the input points to mark them for *replanning, inspection*, etc. When all points are marked, the actions are derived (line 14) using an asynchronous function call. For unplanned points, the configured planners are then invoked to generate the path changes with respect to their planning target (lines 17-24). A list of constraints is shared between all the planners. Finally, the waypoint list Q is updated (line 25).

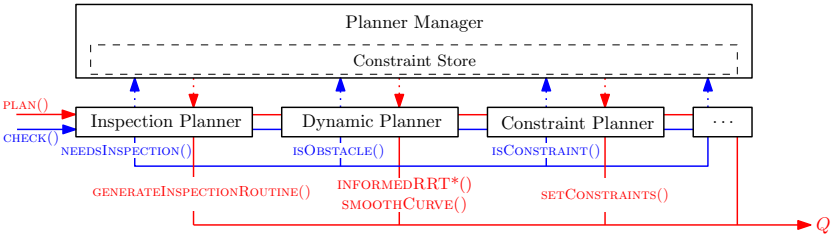


Figure 3.3: Scheme of planner invokations in Planner Manager. Multiple planners can be attached to the Planner Manager, they need to implement *plan* and *check* interfaces. The blue flow illustrates the check functionality for a point. A dynamic Planner would check for free space necessary to approach the point, while a constraint planner would check for user defined constraints preventing this point to be a target. The red flow illustrates the generation of actions to resolve points which need replanning. A dynamic planner might invoke some RRT* routine while an inspection planner generates a custom inspection trajectory.

Since the order of invocation within the Planner Manager is crucial, different priorities can be assigned to the planners. Typically, a low priority planner such as the Inspection Planner should be invoked first, followed by the Dynamic Planner with collision avoidance capabilities. Finally, a Constraint Planner may reject some of the generated points and trigger another iteration of planning.

Algorithm 3.1 Structure of the Planner Controller Routine

Input: Waypoints $\{s_1, \ldots, s_n\}$

$\mathcal{T} \leftarrow$ smooth trajectory through $s_1, \ldots, s_n$ by solving unconstraint QP [11]
$\mathcal{P} \leftarrow$ Manager{ConstraintPlanner, RRTPlanner, InspectionPlanner}
$Q \leftarrow$ sampled waypoints from $\mathcal{T}$ up to horizon h
function UPDATEHORIZON // 5hz rate
    $t \leftarrow$ current time
    Q.prune()
    Q.update($\mathcal{T}(t), \mathcal{T}(t + \Delta), \ldots, \mathcal{T}(t + h\Delta)$)
    if $\mathcal{T}(t) \in$ acceptanceRadius(s_t) **then** s_t.reached $\leftarrow$ **true**
    for all $q \in Q$ with q.state $=$ wp_state::future **do**
        for all $p : \mathcal{P}$ **do**
            q.planning_state $\leftarrow$ p.check(q, t)
        end for
    end for
    generateActions() **async**
end function

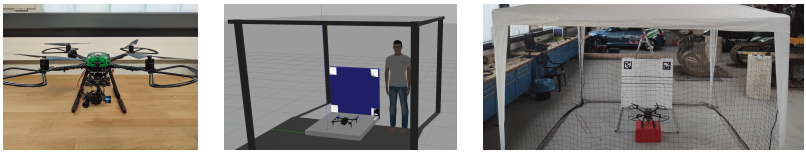
function GENERATEACTIONS
    V $\leftarrow$ Q.rangeView($q =>$ q.state $\neq$ planning_state::planned)
    $\mathcal{C} \leftarrow$ collectConstrains(V)
    for all $p : \mathcal{P}$ **do**
        if V contains unplanned points **then**
            $V_p \leftarrow$ p.plan(Q.begin(), V.begin(), $s_t, \mathcal{C}$)
            V.update(V_p)
        end if
    end for
    Q.update(V)
end function

Output: Collision free setpoints in Q

Safety Actions may be triggered in the *updateHorizon* procedure. The invocation tree of three exemplary planners managed by a common manager is visualized in 3.3.

4 Experimental Evaluation

We verify the functionality of the proposed system in both simulation and practical demonstrations. We develop the pipeline with focus on cross-platform operability in order to support different *companion computers*. We equip both the virtual model and the drone used for the practical experiments with a stereo setup (720px $\times$ 480px per lens) for depth perception as well as a $4k$-RGB sensor and a forward facing distance sensor. In addition, we use a 9-axis IMU (3-axis accelerometer, 3-axis gyroscope and 3-axis compass) as well as barometer. We allow to set user-defined values for IMU noise parameters as well as camera calibration parameters and forward the values to the respective modules.



(a) UAV (b) Simulation (c) Real Environment

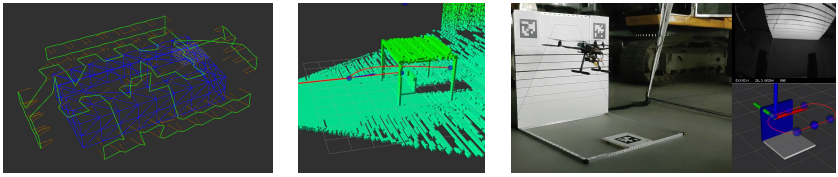
Figure 4.1: Planning tests can be performed in simulation as well as in the real environment.

4.1 In Simulation

We used the px4 *Software-In-The-Loop* component to run experiments with Gazebo as simulator. We recreated the real environment in simulation to test the localization, planning and mapping procedures (cf. 4.1(b)). Even if Gazebo is not capable of rendering photorealistic environments, it allows for a realistic dynamic simulation and connects well with ROS-based system. Sensor and environment data can be read programmatically using standardized interfaces. Checks, such as disabling the planning framework mid-flight, can be performed safely in the simulation environment.

Figure 4.2 shows some of the experiments performed in the simulation environment. In 4.2(a) we calculate inspection points for a simulated building using the static planner module. We tested the dynamic obstacle avoidance

as visualized in 4.2(b). The module conveniently allows to test and evaluate different dynamic planners leveraging the OMPL library. Additional safety checks can be performed to ensure that the replanned trajectory meets specific constraints such as a maximal path length. In the depicted scenario, we used the InformedRRT* planner as default choice. We assumed a $60 \times 60 \times 40\text{cm}^3$ hit volume for the planner's collision check. As admissible heuristic for the planner, we chose the *CostToGoal* heuristic, which is basically the euclidean distance to the target position.



(a) Static Planner

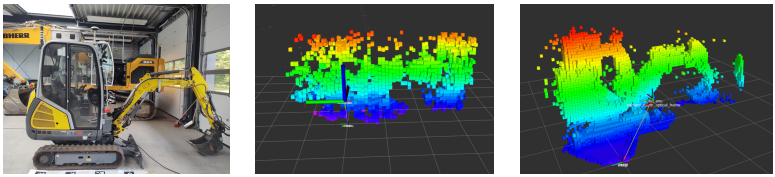
(b) Dynamic Planner in Simulation

(c) Inspection Planner

Figure 4.2: In (a), we invoked a static planning procedure (derived from [1]) resulting in a set of viewpoints. In (b) we can see the result of a dynamic RRT* Planner avoiding an obstacle on the flight path, whereas in (c) an actual short inspection routine is carried out during a real flight.

4.2 On a Real Drone

Figure 4.1(a) shows our hardware and environment setup. We used a standard *Holybro S500* frame with a *Pixhawk 4* as lowlevel flight controller. We designed different custom plug-in mounts to allow for different companion computers and sensor setups. Planning and localization tests have been performed using the Snapdragon Flight module equipped with a quad-core *Snapdragon820* processor using the ARM-v8 architecture. To enable the usage of modern cameras, we also deployed the pipeline onto the *Jetson Xavier NX* as companion computer and connected a *Realsense L515* solid-state LiDAR camera. This setup allows for a more detailed indoor map generation than the stereo setup described above. Figure 4.3 shows a comparison of the two hardware setup in terms of mapping performance.



(a) Scene (b) Stereo Setup (c) Solid-State LiDAR

Figure 4.3: The scene in (a) was captured into a map representation using different sensor setups. (b) shows an exemplary result using a stereo setup while the LiDAR setup used in (c) allows for a more detailed scene representation.

The Localization module (cf. 3.2) allows to run different localization algorithms. Figure 4.4 compares two different approaches. The *Snapdragon Flight* board comes with a basic visual-inertial algorithm based on an Extended Kalman Filter. [8]. We referenced the local pose within the global frame $\mathcal{W}$ by mounting AprilTags [10] at different locations and passing the resulting camera poses in $\mathcal{W}$ back to the EKF. The resulting poses (grey line) are used as baseline as they provide centimeter accurate pose estimates at measured reference points. OrbSlam2 [8] uses only the monocular camera to estimate the pose. The blue line in Figure 4.4 shows that while loop-closure is successfully performed on the right part of the trajectory, on the left part not enough visual features are available to provide an accurate state estimate.

We evaluate the waypoint following capabilities of the pipeline by flying trajectories through manually defined inspection points. Figure 4.2(c) shows an excerpt of such an inspection flight. We notice that the flight performance highly depends on the stability of the localization module. Frequent divergence of the visual inertial system causes the drone to fallback to its safety hovering action preventing a smooth flight.

[7] `https://michaelgrupp.github.io/evo/`
[8] `https://developer.qualcomm.com/software/machine-vision-sdk`

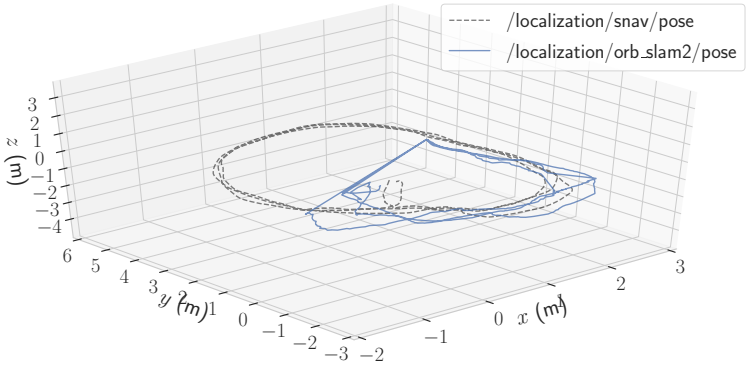


Figure 4.4: Trajectory estimates for repeated circle flight using two different localization systems. The grey line depicts the estimated pose of the reference system, which is a VISLAM approach for the custom *Snapdragon Flight* platform. The tracking camera and the onboard IMU are fused in an EKF fashion to provide a state estimate. In addition, AprilTags have been integrated to define a precise reference frame and enhance the state estimation. The blue line shows the state estimate provided by OrbSlam2 [8] using the monocular tracking camera only. The evaluation can be conveniently performed using EVO [9]as part of the pipeline.

5 Conclusion and Future Work

In this report, a pipeline for performing UAV flights in indoor environments has been introduced. The construction of such a pipeline is motivated by the lack of controlling software for research oriented flight experiments as presented in Section 2. The presented interfaces allow the dynamic usage of different modules for localization, mapping and planning. The required building blocks and their interface definitions have been established, before the cascaded planner structure was presented in detail. Finally, different experiments have been conducted in simulation as well as in real-world scenarios to verify the practicability of the pipeline. The experiments showed that the pipeline is capable to safely perform and operate simple indoor flights.

In order to enhance the robustness of the pipeline further, additional failchecks and fallbacks need to be implemented in future works. This should allow for a

smooth trajectory tracking even with inaccurate localization results. In addition, the dynamic planner module needs additional robustification for application in practice. In the current setup, the feasibility of a planned route is not rechecked after an evasive maneuver. Therefore, such a maneuver can only be performed slowly as the transition to the next planned waypoint may not be smooth. So far, the setup has only been tested with static obstacles. Moving obstacles have higher requirements regarding the real-time capability of the Mapping module which should be investigated in the future.

References

[1] A. Bircher, K. Alexis, M. Burri, P. Oettershagen, S. Omari, T. Mantel and R. Siegwart. "Structural Inspection Path Planning via Iterative Viewpoint Resampling with Application to Aerial Robotics". In: *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. May 2015, pp. 6423–6430.

[2] Tomás Báca et al. "The MRS UAV System: Pushing the Frontiers of Reproducible Research, Real-world Deployment, and Education with Autonomous Unmanned Aerial Vehicles". In: *CoRR* abs/2008.08050 (2020). arXiv: 2008.08050. URL: https://arxiv.org/abs/2008.08050.

[3] M. Burri et al. "Real-time visual-inertial mapping, re-localization and planning onboard MAVs in unknown environments". In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2015, pp. 1872–1878.

[4] Fadri Furrer et al. "Robot Operating System (ROS): The Complete Reference (Volume 1)". In: ed. by Anis Koubaa. Cham: Springer International Publishing, 2016. Chap. RotorS—A Modular Gazebo MAV Simulator Framework, pp. 595–625. ISBN: 978-3-319-26054-9. DOI: 10.1007/978-3-319-26054-9_23. URL: http://dx.doi.org/10.1007/978-3-319-26054-9_23.

[5] Jonathan D. Gammell, Siddhartha S. Srinivasa, and Timothy D. Barfoot. "Informed RRT*: Optimal Incremental Path Planning Focused through an Admissible Ellipsoidal Heuristic". In: *CoRR* abs/1404.2334 (2014). arXiv: 1404.2334. URL: http://arxiv.org/abs/1404.2334.

[6] Armin Hornung et al. "OctoMap: An Efficient Probabilistic 3D Mapping Framework Based on Octrees". In: *Autonomous Robots* (2013). Software available at http://octomap.github.com. DOI: 10.1007/s10514-012-9321-0. URL: http://octomap.github.com.

[7] Taeyoung Lee, Melvin Leok, and N. Harris McClamroch. "Geometric tracking control of a quadrotor UAV on SE(3)". In: *49th IEEE Conference on Decision and Control (CDC)*. 2010, pp. 5420–5425. DOI: 10.1109/CDC.2010.5717652.

[8] Raúl Mur-Artal and Juan D. Tardós. "ORB-SLAM2: an Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras". In: *IEEE Transactions on Robotics* 33.5 (2017), pp. 1255–1262. DOI: 10.1109/TRO.2017.2705103.

[9] Helen Oleynikova et al. "Voxblox: Incremental 3D Euclidean Signed Distance Fields for On-Board MAV Planning". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2017.

[10] Edwin Olson. "AprilTag: A robust and flexible visual fiducial system". In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, May 2011, pp. 3400–3407.

[11] Charles Richter, Adam Bry, and Nicholas Roy. "Polynomial trajectory planning for aggressive quadrotor flight in dense indoor environments". In: *Robotics Research*. Springer, 2016, pp. 649–666.

[12] L. Schmid et al. "An Efficient Sampling-Based Method for Online Informative Path Planning in Unknown Environments". In: *IEEE Robotics and Automation Letters* 5.2 (Apr. 2020), pp. 1500–1507. ISSN: 2377-3774. DOI: 10.1109/LRA.2020.2969191.

[13] V. Usenko et al. "Real-Time Trajectory Replanning for MAVs using Uniform B-splines and a 3D Circular Buffer". In: Vancouver, Canada, Sept. 2017.

[14] Kun Xiao et al. “XTDrone: A Customizable Multi-Rotor UAVs Simulation Platform”. In: *CoRR* abs/2003.09700 (2020). arXiv: 2003.09700. URL: https://arxiv.org/abs/2003.09700.

[15] Boyu Zhou et al. “FUEL: Fast UAV Exploration using Incremental Frontier Structure and Hierarchical Planning”. In: *CoRR* abs/2010.11561 (2020). arXiv: 2010.11561. URL: https://arxiv.org/abs/2010.11561.

A Review on Approaches for Causal Structure Identification

Josephine Rehak

Vision and Fusion Laboratory
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
Josephine.Rehak@kit.edu
ORCiD: 0000-0001-6139-9703

Abstract

Learning the skill to discover causal relations and to make use of them is said to be an essential step in human intelligence [43, 31] and potentially also in machine intelligence [28, 38]. The domain of causal discovery tackles the challenge of identifying causal structures from data collected from observations or experiments by exploiting special properties of causal relations. While current causality literature focuses on methods of probabilistic discovery using conditional independence tests and hard and soft interventions[27, 5, 19] other lesser known approaches are neglected [33].
In this work, we will give a short review on approaches for gaining causal knowledge and provide a categorization of methods. Also, we will introduce the Joint Discovery Assumption that is essential for combining different approaches for causal discovery. Finally, we discuss the open research fields we deduce from our categorization.

1 Introduction

Would it not be great if artificial intelligence (AI) could understand the cause and effects of its past actions and adapt accordingly? What sounds like a dream may be supported by causal-understanding AI in the future. What is called *causal discovery* or *causal structure learning* may level the path for causal reasoning in AI. The strength of these methods lies in finding causal relationships, while also being able to distinguish correlation from causation which is a weakness of many current machine learning methods[38]. For us humans, we are easily able to understand and use causal connections since infancy[22]. The step to causal understanding was a major breakthrough in human evolution [43, 31] the same may be true for the evolution of AI [28, 38].

Currently, such tasks are still beyond the possibilities of an AI as current algorithms fail to keep up with human capabilities. They struggle in simple tasks as when trying to discover the causal connection between the altitude of a weather station above sea level and its average outdoor temperature [15]. The actual causal connection is clear for us humans to see, because we know the average temperature cannot influence the altitude of a weather station.
Our assumption is that future algorithms will have to combine several existing approaches for discovery to gain the most causal structure information. In this paper, we will give a review on such approaches to gain what we call causal structure clues: information about the presence or absence of causal relations in a causal graph, by considering research in various science domains. We contribute by:

- categorizing algorithms to gain causal structure clues.
- providing a common ground for joint inference over various science domains by the definition of the Joint Discovery Assumption.
- deriving prospects for future research.

We cover methods used by humans, but also more advanced methods based on probabilities and spacetime which commonly challenge human understanding. We do not investigate methods for identifying whole causal networks from the

given structure clues as this would exceed the scope of this work.
In section 2, we give a short overview on the fundamentals on general causal structure learning. In section 3, we introduce a fundamental assumptions for the joint use of multiple approaches for causal discovery. The categorization itself is explained in section 4. We provide a short analysis of future prospects based on made categorization in section 5 and summarize our findings in section 6.

2 Basics of Causal Structure Discovery

In causal discovery, one tries to fully discover the true causal graph G^* for a given process under investigation. Such a graph consists of a set of variables V depicted as nodes and a set of edges E connecting the variables. It is a common assumption that G^* is always a causal directed acyclic graph (DAG); causal, as all edges in E indicate causal relations between the variables in V; directed, as all edges in E are only allowed to be only one-directed or absent; acyclic, as the edges in E may not form any cycles like bi-directed relations. Cycles may only occur in temporal considerations.
For given structure knowledge over the absence or direction of edges of G^*, a set of equally possible DAGs can be inferred which form an equivalence class of DAGs with respect to the provided knowledge. These equivalence classes can be represented by Partially Directed Acyclic Graphs (PDAGS). If no existing knowledge is present, the equivalence class contains all feasible DAGs for given variables. For this case, the number of DAGs is described in [41]. By adding structure knowledge to an equivalence class, one may direct or eliminate edges and thereby restrict the equivalence class. G^* is deemed to be fully discovered if all possible edges between the variables of V are discovered to be either absent or one-directed [27].
Approaches to gain such causal structure knowledge are explained in section 4 in detail.

3 Joint Discovery Assumption

We humans naturally use multiple approaches to identify causal relationships. This is based on the assumption that no matter which causal discovery method is used the underlying graph is the same as long as no undetected changes have occurred in the graph. As long as each structural clue is implied by the causal graph the discovered causal clues cannot contradict each other unless when essential assumptions of the discovery methods were disregarded.
As an example, if two causal discovery methods m_1, m_2 infer a causal relation between the variables A and B results in two structural clues c_1 and c_2, then c_1 must be supported by c_2 and c_2 must be supported by c_1 accordingly. This means the discoveries $c_1 = \{A \to B\}$ and $c_2 = \{A \perp\!\!\!\perp B\}$ are not be possible without one of the clues to be erroneous. But, for example $c_1 = \{A \to B\}$ and $c_2 = \{A \not\perp\!\!\!\perp B\}$ are two structural clues that fully support each other.
We call this Joint Discovery Assumption, since it allows the joint use of several discovery methods. Based on this assumption, we can combine various structure clues from data to gain the most information on the true causal graph and we can evaluate more different types of data, since results from matching discovery algorithms can be included in the discovery process. In addition, computationally inexpensive methods can be used to restrict computationally or cost intensive discovery methods. Such an application could lead to a considerable streamlining, since previously excluded causal relationships no longer need to be considered by the second method. Note however that not every causal discovery method could infer every structural clue, since each comes with its own presuppositions.

4 Approaches for Discovering Causal relations

In the following, we grouped methods by similarity in their approach of exploiting properties of causal relations for inferring causal structure clues.

4.1 Structure clues from expert knowledge

The easiest discovery approach for artificial intelligence to learn causal relations is to get them taught by humans. We define such a structure clue as any knowledge that can be used for the construction of causal graphs. This may include knowledge over specific known edges [24], a known causal order [9], a known partial causal order [36], a known path between variables [3] or even knowledge over variable types [4]. We will deal with the latter in particular.
Typing assumptions [4] can only be applied if the investigated variables contain variables of similar type, e.g. multiple diseases or multiple drugs for treatment, and if structure knowledge over the type in relation to other variables is given, e.g. variables of type 'disease' may cause variables of type 'symptoms'. Such a typification of variables has to be performed by domain experts in advance. The general background knowledge over types can be used to limit the causal structure search. For example the provided domain knowledge may entail that diseases may cause symptoms, but symptoms may not cause diseases. The causal discovery methods can then consider these general rules in their detailed search. This example requires manual typification of the variables though investigation in automated type classification of variables may deliver promising results [4]. Using expert knowledge in general is a comfortable way of combining prior domain knowledge with the power of discovery algorithms. It is also an easy way to speed up discovery by reducing the amount of edges that need to be considered in causal structure search while leaving the actual causal discovery task to an algorithm. Unfortunately, introducing expert knowledge comes with the risk of introducing structure faults which can result in a wrong causal graph.

4.2 Structure clues from probabilistic inference

Since Judea Pearl published the foundations for probabilistic causal discoveries in 1988, the domain of probabilistic discovery is flourishing. By collecting data via observations with additional knowledge over the variable states, we are able to recover probabilities. Depending on the type of algorithm, these are interpreted differently. In the domain, the categorization into score-based and constraint-based algorithms has become established [11]:

Constraint-based methods make use of patterns in the retrieved probabilities to uncover fragments of the causal structure. For one, unconditional independence tests allow the learning of skeletons by identifying the presence, but not the direction, of causal relations. Further on, conditional independence tests can uncover immoralities, sometimes also called uncovered colliders or v-structures, by making use of d-separation properties [29]. Score-based algorithms try to find the graph with the highest fit to the data by varying edges in the graph. The accuracy of fit is measured in a likelihood score as the Bayesian information criterion [2] or the Aikaike information criterion [1]. A common example is the Greedy Equivalence Search (GES) [7].
The third category forms a group of mixture methods that use both score-based and constraint-based learning like Max-Min Hill Climbing (MMHC) [45].
All these approaches can be used to learn Markov Equivalence Classes (MEC), equivalence classes of DAGs which are equivalent in their conditional and unconditional independencies [46]. MECs can be represented in the form of complete partially directed acyclic graphs (CPDAGs) [42]. For each algorithm the resulting MECs may differ depending on found conditional independencies. Unfortunately, MECs can still include countless DAGs, especially when investigating big data [14]. Hence, the current challenge in this domain lies within finding the structure knowledge to reduce the MEC further. For this purpose, some publications resort to other discovery approaches to orient the remaining edges for example by using interventions [10] (see section 4.3) or noise methods [26] (see section 4.5). All in all, using probabilistic discovery we can discover important elements of the true causal graph, but may not discover it completely. This comes at the disadvantage that the discovery relies completely on the availability of huge quantities of unbiased data. Without such large sample sets, it is hard to gain profound results from (un)conditional independence tests [29].

4.3 Structure clues from manipulation

Another approach to discover causal relations that comes natural for humans is by experimenting with manipulations of variables. With a chosen intervention, a variables state or probability of occurrence is changed. This may trigger a change in the causally dependent variables which can be measured and

allows conclusions about causal dependencies. This builds on the fundamental assumption that an intervention targeted on the cause may influence the effect, but an intervention targeted at the effect may never create a change in the cause. An essential aide for such experiments are ceteris paribus conditions where the environmental variables can be recreated so several interventions may be tried out. These interventions are commonly of the following two types: Hard interventions, also called Pearlian interventions or structure interventions, forcefully set a variable to a chosen value and thereby eliminate all influences from other causes. They were first formalized in Judea Pearls do-calculus [30]. This calculus was proven to also fully support the popular potential outcomes framework [34].
So called soft interventions introduce an additional variable which causally affects the target variable and changes its probability of occurrence without disrupting the influence of other variables [5, 19]. For both kinds, the numbers of interventions required for the full identification of a true causal graph were identified by [10]. Also, several methods were established to identify the effects created by these interventions. Assuming strong ceteris paribus conditions, the effect of structure interventions for example can be discovered and measured by the Average Treatment Effect (ATE). It is calculated as the normalized sum over the individual treatment effects of all individuals or samples $i = 1, ..., N$. The individual treatment effect is the difference of the treated outcome variable $y_1(i)$ and the untreated individual outcome variable $y_0(i)$.

$$\text{ATE} = \frac{1}{N} \sum_i (y_1(i) - y_0(i))$$

In simple scenarios, few samples of each interventional outcome may suffice to identify the causal effect of interventions to allow structure deductions.
The ATE can also be estimated which is useful when interventions can be observed but not preferably applied. Common methods are the difference in difference methods [37], propensity score matching [35], and regression discontinuity designs [16].
Additionally, interventional effects can be identified by probabilistic discovery methods described in 4.2. Commonly, structure interventions can be identified with unconditional independence tests as the graph skeleton is disrupted. While soft interventions can be identified by conditional independence tests as by

adding new causes to a variable a new v-structure is created.
All in all, discovery by manipulation is a less favored discovery approach as interventions can be costly, unethical or even unfeasible for some variables. Also the strong assumption of ceteris paribus conditions are a disadvantage since some conditions are hard to reciprocate and can most often only be tackled by averaging over larger sample sets.

4.4 Structure clues from functional modeling

This approach uncovers causal relations by remodeling how the effect emerged from the cause as the cause needs to produce the effect.
So called functional causal models consist of a causal graph and a set of functions that relate the variables of the graph in accordance to the graph edges. An approach of structure discovery is to uncover the causal graph by retracing the causal functions. For example, Linear, Non-Gaussian, Acyclic causal Models (LiNGAM) [39] is an algorithm that uncovers a multivariate DAG structure by computing linear functions of the variables X and a connection strength matrix $\mathbf{B}$ plus additive noise ϵ, $X = \mathbf{B} * X + \epsilon$, by the use of an independent component analysis [17, 8]. Other methods that follow a similar approach are Additive Noise modeling, or Post Nonlinear Causal Model described in the next subsection.
These methods do not require the faithfulness assumption, but assume causal sufficiency: the absence of confounding variables. They also require the assumption that the additive noise are non-Gaussian distributions of non-zero variances, also abbreviated as *non-Gaussianity assumption*, because [42] has shown that methods that use only covariance matrices have no way of inferring the direction of the causal relation. Another fallacy of this approach is that it is prone to spurious correlations. Those may equally result in believable functions, but are not causally related. Hence the combination with another approach is strongly advised.

4.5 Structure clues from noise

To discover causal relations by the use of noise is a rather new notion. The fundamental property of causal relationships is exploited that natural noise found in the cause needs also to be found in the effect, but no noise of the effect may be found in the cause. This approach came up with Additive Noise Modeling (**ANM**), originally a functional discovery approach which also makes use of the noise property to gain certainty of the relation to be causal. Other than LiNGAM, the non-linear function $Y = f(X) + \epsilon$ of two variables X and Y is reconstructed from data. First, a regression for $X \to Y$ and $Y \to X$ is performed to approximate the relationship function f, then the residual ϵ is calculated and finally, the residuals are tested for independency [25]. This method has shown to be prone to confounding and feedback noise as both mess with the aforementioned noise property. An extension to ANMs are Post Nonlinear Causal Models (PNLs) [48] which also take nonlinear distortions f_2 from sensor or measurement errors into account by recovering $Y = f_2(f_1(X) + \epsilon)$.
Naturally, the approach of using the noise property relies on the presence of noise which is not always given. Also, it requires a large sample size of observations to reliably retrace the causal function.

4.6 Structure clues from time, space and spacetime information

Temporal and spatial information have shown to be the most important cues for human causal understanding [21]. For humans the temporal sequences of events are particularly important for the discovery of causal relationships. Events that follow a chosen event are often understood as consequences. Whereas events that precede it are understood as causes of that event [6]. Hidden in this understanding is the basic, well-known assumption that in time, a cause must always precede its effect.
Another early notion of causal understanding is the spatial proximity of the effect to its causes. In physics, this notion was called principle of locality: two objects may only be causally connected by mechanical influences as for example by touch. With the discovery of electromagnetic and gravitational waves this

notion changed. By today, we know causal relations are not only bound to space or to time but to spacetime as the travel of a causal signal is fundamentally limited by lightspeed. More current research inspects causal relations in relation to algebra in four-dimensional Minkowski spacetime M of special relativity as it can not be represented in Euclidean geometry [44]. Therein, a set of points form a region in M. An event at point x emits two light cones: a *forward lightcone* $V_+(x)$ emitted into the future, and a *backward lightcone* $V_-(x)$ emitted into the past. A point y of an event caused by x has per definition to be in $V_+(x)$, while a point of an event causing x has to be in $V_-(x)$. Any event caused by x and y has to lie within $V_+(x) \cap V_+(y)$.
The result of an intersection of a forward and a backward cone is called a *double cone*. Each double cone is causally complete, bounded, closed, and convex. The new law of locality can be derived from it: two regions in M are causally disjoint and thereby physically independent if they are spacewise separated [44]. As measurements of events in Minkowski spacetime tend to be imprecise, current literature also tackles the implications of imprecise time and location measurements and time-frame measurements [20].
The consideration of regions in Minkowski spacetime may add to the considerations in causal discovery. For example, [47] created a theoretical framework for causal image synthesis using knowledge over Minkowski spacetime.

4.7 Structure clues from forecasting and prediction

This approach assumes that if two variables are causally connected then we should be able to predict the effect given the cause. It is closely related to functional modeling, but differs in the fact that not the causing function is recovered, but instead we investigate how the prediction improves, if we add or remove knowledge over the potential cause. This approach is especially popular in timeseries. The earliest method was Granger causality for applications in the economy [13]. A stationary timeseries X is said to granger-cause another stationary timeseries Y considering X when calculating the variance of the residual of predicting Y creates a noticable change. This does not include any definition of the predicting function itself.

A derivative of Granger causality is Instantaneous causality [32] which defines X and Y as instantaneously causally related if adding the value X_i for timepoint i improves the prediction of Y_i. Countless other methods have developed in this domain like Sims causality [40] or multistep causality [23]. A common shared weakness of these methods is a sensitivity to feedback or confounders, latent variables that actually cause the observed variables, since these also allow to make predictions but are not based on a direct causal relationship.

5 Future Prospects

Future causal discovery methods will be highly dependent on the availability of data, but also on interpreting it most efficiently. Throughout the course of this paper, we highlighted several causal approaches to gain structure knowledge from all kinds of data to construct causal graphs on. We assume that efficient causal discovery algorithms will have to apply several approaches as the discovery potential of each approach is limited. With this combination of approaches, new research questions arise: 1) As each new approach is able to reduce the equivalence class of DAGs new types of equivalence classes come up which are in comparison to Markov equivalence classes of probabilistic discovery heavily underexplored. 2) New combinations of the approaches are possible which have not been investigated yet, as for example discovery methods using interventions to artificially introduce noise for discovery. For some combinations of approaches, the foundation stone is laid but still require additional work, like implementations of probabilistic algorithms that can use existing domain or expert knowledge. 3) We see a prospect in a new kind of active causal structure learning that can apply each approach most cost-efficiently for structure learning to have the highest knowledge gain [12]. 4) Common applications of causal discovery do not include technical systems, but the methods show high potential for this domain, as experiments on machines cannot be unethical and states can be easier intervened on and reciprocated [18].

6 Small Overview

We introduced the Joint Discovery Assumption which allows the joint use of multiple causal discovery methods. Also, we gave a slim overview over various notions to gain causal structure clues, i.e. the presence or absence of causal relations. For each approach, advantages, assumptions, and limits were identified. For some of the methods listed, we need additional information as temporal and or location data to make causal deductions. While in the case of expert knowledge or additional domain knowledge, the learning process requires human assistance.
Some approaches, as structure clues from spacetime, provide only structure clues over the absence of edges. This shows potential to be a cheap possibility of using additional information, while speeding up the structure search algorithms by eliminating invalid spurious relations in advance. Finally, we made a basic proposal for further research based on the presented approaches.

References

[1] Hirotogu Akaike. “Information theory and an extension of the maximum likelihood principle”. In: *Selected papers of hirotugu akaike*. Springer, 1998, pp. 199–213.

[2] Harish S Bhat and Nitesh Kumar. “On the derivation of the bayesian information criterion”. In: *School of Natural Sciences, University of California* 99 (2010).

[3] Giorgos Borboudakis and Ioannis Tsamardinos. “Incorporating Causal Prior Knowledge as Path-Constraints in Bayesian Networks and Maximal Ancestral Graphs”. In: *International Conference on Machine Learning* (2012).

[4] Philippe Brouillard et al. “Typing assumptions improve identification in causal discovery”. In: *International Conference on Machine Learning Workshop* (2021).

[5] John Campbell. “An interventionist approach to causation in psychology”. In: *Causal learning: Psychology, philosophy and computation* (2007), pp. 58–66.

[6] Patricia W Cheng and Laura R Novick. “A probabilistic contrast model of causal induction.” In: *Journal of personality and social psychology* 58.4 (1990), p. 545.

[7] David Maxwell Chickering. “Optimal structure identification with greedy search”. In: *Journal of machine learning research* 3 (2002), pp. 507–554.

[8] Pierre Comon. “Independent component analysis, a new concept?” In: *Signal processing* 36.3 (1994), pp. 287–314.

[9] Gregory F Cooper and Edward Herskovits. “A Bayesian method for the induction of probabilistic networks from data”. In: *Machine learning* 9.4 (1992), pp. 309–347.

[10] Frederick Eberhardt and Richard Scheines. “Interventions and causal inference”. In: *Philosophy of science* 74.5 (2007), pp. 981–995.

[11] Clark Glymour, Kun Zhang, and Peter Spirtes. “Review of Causal Discovery Methods Based on Graphical Models”. In: *Frontiers in Genetics* 10 (2019), p. 524. ISSN: 1664-8021. DOI: 10.3389/fgene.2019.00524.

[12] Julius Gonsior et al. “Active Learning for Spreadsheet Cell Classification.” In: *EDBT/ICDT Workshops*. 2020.

[13] Clive WJ Granger. “Investigating causal relations by econometric models and cross-spectral methods”. In: *Econometrica: journal of the Econometric Society* (1969), pp. 424–438.

[14] Yangbo He, Jinzhu Jia, and Bin Yu. “Counting and exploring sizes of Markov equivalence classes of directed acyclic graphs”. In: *The Journal of Machine Learning Research* 16.1 (2015), pp. 2589–2609.

[15] Patrik Hoyer et al. “Nonlinear causal discovery with additive noise models”. In: *Advances in Neural Information Processing Systems*. Vol. 21. Curran Associates, Inc., 2009, pp. 689–696.

[16] Guido W Imbens and Thomas Lemieux. “Regression discontinuity designs: A guide to practice”. In: *Journal of econometrics* 142.2 (2008), pp. 615–635.

[17] Christian Jutten and Jeanny Herault. “Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture”. In: *Signal processing* 24.1 (1991), pp. 1–10.

[18] Andreas Kimmig et al. “Wertstromkinematik–Produktionssysteme neu gedacht”. In: *Zeitschrift für wirtschaftlichen Fabrikbetrieb* 116.12 (2021), pp. 935–939.

[19] Kevin Korb et al. “Varieties of causal intervention”. In: *Pacific Rim International Conference on Artificial Intelligence* 3157 (2004), pp. 322–331. DOI: 10.1007/978-3-540-28633-2_35.

[20] Olga Kosheleva and Vladik Kreinovich. “Observable causality implies Lorentz group: Alexandrov-Zeeman-type theorem for space-time regions”. In: 2 (30) (2014).

[21] David A Lagnado and Steven A Sloman. “Time as a guide to cause.” In: *Journal of Experimental Psychology: Learning, Memory, and Cognition* 32.3 (2006), p. 451.

[22] Alan M. Leslie and Stephanie Keeble. “Do six-month-old infants perceive causality?” In: *Cognition* 25.3 (1987), pp. 265–288. ISSN: 0010-0277. DOI: https://doi.org/10.1016/S0010-0277(87)80006-9.

[23] Helmut Lütkepohl and Maike M Müller. “Testing for multi-step causality in time series”. In: (1994).

[24] Christopher Meek. “Causal inference and causal explanation with background knowledge”. In: *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*. 1995, pp. 403–410.

[25] Joris Mooij and Dominik Janzing. “Distinguishing between cause and effect”. In: *Causality: Objectives and Assessment*. PMLR. 2010, pp. 147–156.

[26] Pramod Kumar Parida, Tshilidzi Marwala, and Snehashish Chakraverty. “A multivariate additive noise model for complete causal discovery”. In: *Neural Networks* 103 (2018), pp. 44–54.

[27] Judea Pearl. “Causal diagrams for empirical research”. In: *Biometrika* 82 (1995).

[28] Judea Pearl. *Causality*. Cambridge university press, 20090. DOI: 10.1017/CBO9780511803161.

[29] Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan kaufmann, 1988.

[30] Judea Pearl. *The Do-Calculus Revisited*. 2012. arXiv: 1210.4852 [cs.AI].

[31] Daniel J Povinelli and Jesse M Bering. "The mentality of apes revisited". In: *Current Directions in Psychological Science* 11.4 (2002), pp. 115–119.

[32] J Michael Price. "The characterization of instantaneous causality: A correction". In: *Journal of Econometrics* 10.2 (1979), pp. 253–256.

[33] Josephine Rehak. "A Proposal on Discovering Causal Structures in Technical Systems by Means of Interventions". In: 2020, pp. 91–106.

[34] Thomas S Richardson and James M Robins. "Single world intervention graphs (SWIGs): A unification of the counterfactual and graphical approaches to causality". In: *Center for the Statistics and the Social Sciences, University of Washington Series.* 128.30 (2013), p. 2013.

[35] Paul R Rosenbaum and Donald B Rubin. "The central role of the propensity score in observational studies for causal effects". In: *Biometrika* 70.1 (1983), pp. 41–55.

[36] Richard Scheines et al. "The TETRAD project: Constraint based aids to causal model specification". In: *Multivariate Behavioral Research* 33.1 (1998), pp. 65–117.

[37] Hannes Schellhorn. *Effizienzeffekte der Einkommensteuer bei Steuervermeidung*. Springer-Verlag, 2005.

[38] Bernhard Schölkopf. "Causality for machine learning". In: *arXiv preprint arXiv:1911.10500* (2019).

[39] Shohei Shimizu et al. "A linear non-Gaussian acyclic model for causal discovery." In: *Journal of Machine Learning Research* 7.10 (2006).

[40] Christopher A Sims. "Money, income, and causality". In: *The American economic review* 62.4 (1972), pp. 540–552.

[41] N Sloane. *Number of acyclic digraphs (or DAGs) with n labeled nodes.* `http://oeis.org/A003024`. Accessed: 2021-11-02. 2021.

[42] Peter Spirtes et al. *Causation, prediction, and search*. MIT press, 2000. DOI: 10.1007/978-1-4612-2748-9.

[43] Martin Stuart-Fox. "The origins of causal cognition in early hominins". In: *Biology & Philosophy* 30.2 (2015), pp. 247–266.

[44] Launey J Thomas III and Eyvind H Wichmann. "On the causal structure of Minkowski spacetime". In: *Journal of Mathematical Physics* 38.10 (1997), pp. 5044–5086.

[45] Ioannis Tsamardinos, Laura E Brown, and Constantin F Aliferis. "The max-min hill-climbing Bayesian network structure learning algorithm". In: *Machine learning* 65.1 (2006), pp. 31–78.

[46] Thomas Verma, Judea Pearl, et al. "Equivalence and synthesis of causal models". In: (1991).

[47] Athanasios Vlontzos et al. "Causal Future Prediction in a Minkowski Space-Time". In: *arXiv preprint arXiv:2008.09154* (2020).

[48] Kun Zhang and Aapo Hyvärinen. "Distinguishing causes from effects using nonlinear acyclic causal models". In: *Causality: Objectives and Assessment*. PMLR. 2010, pp. 157–164.

Potentials of Spectral Imaging for Stress Monitoring in Viticulture

Petra Schumacher

Vision and Fusion Laboratory
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
petra.schumacher@kit.edu

Abstract

Remote sensing technologies are widely used to monitor quantity and quality of plants in agriculture and forestry. While conventional panchromatic and RGB cameras can provide spatial information equivalent to human vision, high-resolution spectroscopy can reveal information about the chemical and structural composition of plants and provide detailed insight about plant health. With the development of commercial high-resolution multi- and hyperspectral cameras in the past decade, it is now possible to combine spectral and spatial information to obtain high-throughput and accurate predictions of plant condition. In this article, the underlying optical phenomena in plants will be reviewed and projected to potential imaging applications for plant stress monitoring with a focus on viticulture.

1 Introduction

As world population continuously grows and agriculture is increasingly under pressure of climate change, precisely monitoring the state of crops is getting more important. Imaging technologies have become an invaluable tool for

plant phenotyping, i.e. the recording of all visible characteristics of a plant, which result from the interaction of the genetic information and environmental influences. RGB-cameras have been widely available for decades and have proven to be suitable for a large range of applications, including prediction of shoot biomass [19], assessment of disease severity [41] and monitoring of leaf injury [27]. However, the low spectral resolution of RGB-images does not allow detailed conclusions to be drawn about the chemical composition of the plants under observation and, in many cases, does not reveal the cause of specific symptoms. In grapevine, several conditions can result in similar symptoms that can be hard to distinguish by the human eye or RGB-imaging. To overcome this limitation, multi- and hyperspectral cameras have become increasingly powerful tools for plant phenotyping, providing high resolution spectra as well as spatial information. Typically, the investigated spectral range is extended into the near infrared (NIR)- or even short-wave infrared (SWIR) regime, allowing predictions about the chemical composition. A widely-known approach for plant state monitoring is given by the normalized difference vegetation index (NDVI), first introduced in 1973 [37]. It has been derived from satellite imaging bands and describes the ratio of the difference of the signal measured by the red and NIR band, given by g_{red} and g_{NIR}, over their sum

$$NDVI = \frac{g_{\mathrm{NIR}} - g_{\mathrm{red}}}{g_{\mathrm{NIR}} + g_{\mathrm{red}}} \,. \tag{1.1}$$

In the first publication, band 5 and 7 of Landsat 1 were utilized [37], corresponding to 600-700 and 800-1100 nm, respectively. Until today, it has remained one of the most widely used indices to predict general plant condition [18].The broad validity of the NDVI, not only in agriculture, is based on the optical properties of the leaf pigment chlorophyll and the structure of leaves. The influence of the three main pigments in leaves, chlorophyll, carotenoids and anthocyanins as well as the influence of leaf cellular structure will be reviewed in more detail in the following. In Section 3, the influence of plant stress on the optical properties will be described. The technologies and approaches to exploit these properties for stress identification are introduced in Section 4. The article concludes with an outlook on open research questions and future work.

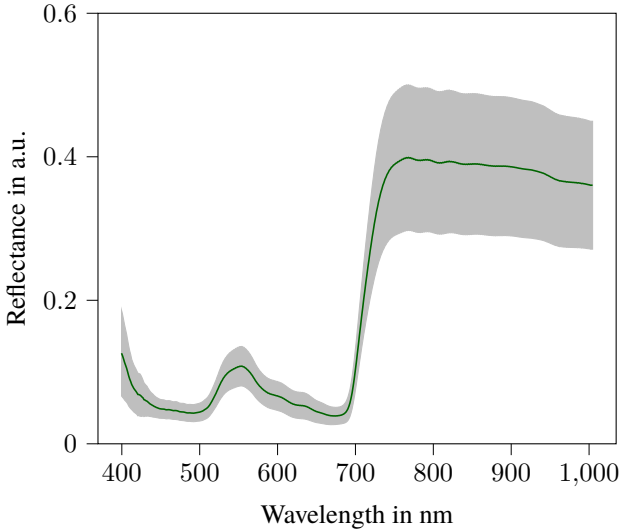


Figure 2.1: Average reflection spectrum measured on a leaf of *Vitis vinifera* L., cultivar 'Riesling' (green line). The shaded area illustrates the standard deviation of all pixels across the whole leaf.

2 Optical Properties of Leaves

As photosynthesis takes place in the leaves, their state is of vital importance for all green plants. The optical properties of leaves can provide valuable insights into potential stress factors affecting the plant, which shall be illustrated in the following on the example of grapevine leaves. An exemplary spectrum of a healthy grapevine leaf of cultivar 'Riesling' is shown in Figure 2.1.

2.1 Optical Phenomena in Leaves

Interaction of light with leaves is governed by surface reflection, internal reflection, light absorption by pigments and transmission through the leaf tissue [43]. Light reflected from leaf surfaces is specularly reflected and typically polarized [43]. This portion of reflected light is not influenced by pigments or

the internal leaf structure. Depending on the surface structure, i.e. the amount and shape of surface wax as well as the number of hair, surface reflectivity varies and can in some species account for most of the reflected light [45, 20]. Light penetrating through the leaf surface is reflected from each cell wall-air interface of the epidermal cells along the optical path due to the higher refractive index of water-filled cells than air-filled spacings between cells [43, 8]. When leaving the leaf after multiple internal reflections, this portion of light is unpolarized. It could be demonstrated that soaking a leaf in water will fill enclosed air gaps and consequently reduce the difference in refractive indices and thus significantly reduce internal reflection in the NIR spectral range [8]. Light propagating inside the leaf tissue is selectively absorbed by pigments, resulting in strongly wavelength-dependent internal reflection depending on the concentration of individual pigments. The characteristic absorption properties of the most prominent pigments are evaluated in the following sections.

2.2 The Sieve Effect and Detour Effect

It is well known from Lambert-Beer's law that absorbance linearly increase with increasing concentration of the absorbing species. However, this relation only holds for homogeneously distributed particles. This is not the case in plant leaves, where particles are not freely floating but are concentrated within cell organelles [25]. Thus, the absorbance of pigments comprised in organelles is smaller compared to the same concentration of pigments homogeneously suspended in the surrounding medium, which is known as the sieve effect and leads to underestimated pigment concentration from absorption spectra. A decrease of the absorption maximum as well as broadening of absorption bands at increased chlorophyll concentration due to the sieve effect has been observed by [8]. On the contrary, abundant air-cell interfaces result in light scattering and thus increase the optical path length within leaves, increasing the chance of light interaction with pigments. This is known as detour effect and, in return, increases absorption and overestimates pigment concentration [43]. In the near infrared regime where chlorophyll does not absorb, scattering results in very high leaf reflectivity. With increasing age of the plant, the intercellular air spaces become more abundant, resulting in increased scattering and thus increased

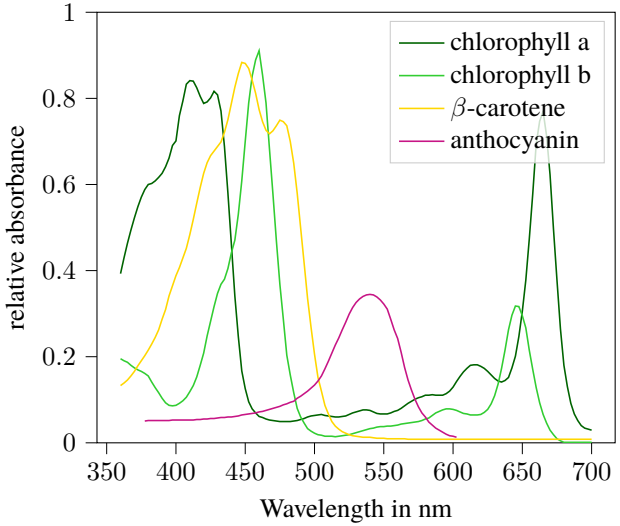


Figure 2.2: Schematic absorption spectra of chlorophyll a (dark green), chlorophyll b (light green), β-carotene (yellow) and anthocyanin (purple). Reproduced from [16].

reflectivity in the near infrared [8]. The relation of sieve effect and detour effect is strongly tissue-dependent and thus not generally describable [25].

2.3 Molecular Absorption of Leaf Pigments

2.3.1 Chlorophyll

The most prominent pigment in plants is chlorophyll, responsible for the green appearance of leaves. Different types of chlorophyll exist that can be distinguished by their side chains. In plants, chlorophyll a and b are present. Their absorption spectra are shown in Figure 2.2. Comparing the absorption spectra to the reflectance spectrum in Figure 2.1, the reflectance dips in the blue and red spectral regions and the green appearance of the leaf can be readily explained. Chlorophyll a strongly absorbs in the blue and red regime, while

chlorophyll b presents a prominent absorption peak in the blue-green and a smaller peak in the orange regime. While chlorophyll a is used as the primary light absorbing molecule in photosynthesis, chlorophyll b is supplemented to enlarge the absorption range and is thus relatively more abundant in shade plants [6]. In grapevine, the ratio of chlorophyll a/b typically ranges between 2 and 3 [28].

The combination of chlorophyll absorption and strong NIR-scattering results in the red edge, the sharp increase of reflectivity in the far red, which can be seen in Figure 2.1 around 700 nm. This effect is the result of two phenomena: firstly, chlorophyll absorption sharply decreases towards the red edge as can be seen in Figure 2.2, resulting in increased reflectivity. Secondly, it can be attributed to strong internal scattering at the leaf cell boundaries in the infrared regime causing strong reflectivity [24]. As shown above, the NDVI describes the relation between red and NIR reflectance and thus a normalized ratio of both sides of the red edge. As the red edge is invariant to background coverage [24], it has become a valuable tool for remote sensing with numerous applications.

2.3.2 Anthocyanins

Anthocyanins are a subgroup of flavonoids, secondary plant metabolites that are responsible for pigmentation [17]. Flavonoids fulfill numerous important tasks for the protection of plants, thus their synthesis rate can be triggered by stress [35]. Anthocyanins encompass a large group of molecules responsible for characteristic red, blue, purple or black coloring of plants.

Malvidin and in particular malvidin-3-O-glucoside has been determined as the most abundant anthocyanin in grapevine [12]. It should be noted that anthocyanin absorption is strongly pH-dependent [46] and the representation in Figure 2.2 is thus only illustrative without general validity.

In red grapevine varieties, anthocyanins are responsible for the purple or red coloring of grapes, wine and sometimes leaves, whereby the concentration is strongly dependent on the cultivar [12]. While the genes responsible for anthocyanin generation are also present in white cultivars, their synthesis is inhibited by mutant regulatory genes [44].

2.3.3 Carotenoids

Carotenoids appear yellow and are molecules involved in the photosynthetic process just like chlorophyll. They consist of liposoluble tetrapenes and can be subdivided into two main classes due to their chemical structure: carotenes, represented by a specific group of hydrocarbons; and xantophylls, respresented by oxygenated carotenes [40]. β-carotene is the most abundant carotene and lutein the most abundant xantophyll in grapevine [23]. Carotenes fulfill two main functions in plant photosynthesis: photoprotection by dissipating excess heat as well as a possible contribution to light harvesting by absorbing light in the green spectral range which is not efficiently absorbed by chlorophyll [15, 22]. Carotenoids are present in leaves all season long, but are typically masked by chlorophyll. With declining chlorophyll concentration in autumn, the influence of carotenoids becomes visible with leaves turning to orange and yellow hues [3].

3 Influence of Stress on Plant Spectra

Pigments are subject to associated metabolic pathways and thus susceptible to environmental influence. Therefore, their concentration can change due to biotic or abiotic stress, resulting in altered reflectance and absorption spectra. Additionally, the cell structure may change due to external influence, which in turn influences leaf scattering properties. The individual effects and their impact on the reflectance spectra of grapevine leaves are explained in the following section.

3.1 Change in Pigment Concentration

3.1.1 Chlorophyll and Carotenoids

As chlorophyll and carotenoids are both involved in photosynthesis, the rate of change of both pigments is often highly correlated. Upon stress exposure, leaf chlorophyll content as well as the concentration of most carotenoids typically

declines. This has been reported for several biotic conditions in grapevine, including plants infected with the bacterial infection Bois Noir [38] and the virus infection Grapevine Leafroll disease GLRaV-3 [21]. It was found that Esca, a fungal disease, results in strongly decreased chlorophyll content, whereas the carotenoid content did not change significantly [34].

The influence of different abiotic stress factors on the concentration of chlorophyll and carotenoid was investigated by [10] on two different red cultivars, 'Touriga Nacional' (TN) and 'Trincadeira' (TR). They observed very different reaction of cultivars upon abiotic stress: while water stress, light stress and heat stress individually resulted in an increase in total chlorophyll concentration in TN, it slightly decreased in TR when compared to a non-stressed control group. Combined light, heat and water stress, in return, resulted in the increase of chlorophyll content in both cultivars. Carotenoid content slightly decreased during exposure to water, light and heat stress for TN, but slightly increased in TR in response to water- and light stress. Combination of water, light and heat stress resulted in increased carotenoid content in TN but was not significantly affected in TR.

3.1.2 Anthocyanins

The change of anthocyanin concentration has been observed as response to a wide range of biotic and abiotic stresse in grapevine, but is decoupled from chlorophyll and carotenoid concentration change. Several studies showed an increase of anthocyanin concentration in red cultivars as response to biotic stress. [21] could identify the presence of anthocyanins only in Grapevine Leafroll disease GLRaV-3 infected leaves, while the healthy control group did not contain any measurable amount of anthocyanin. The response of the two red cultivars 'Nebbiolo' and 'Barbera' to the bacterial infection *Flavescence dorée* (FD) were measured by [31]. 'Barbera', being highly susceptible to FD, showed an increase of anthocyanin by up to ten times compared to the healthy control group, whereas in the less susceptible cultivar 'Barbera', anthocyanin concentration doubled at maximum. Upon recovery, both plants did not exhibit remarkable difference compared to the control group.

The influence of abiotic stress on the cultivars 'Touriga Nacional' (TN) and 'Trincadeira' (TR) on anthocyanin concentration was investigated by [10]. In their experiments, water, light and heat stress individually decreased anthocyanin concentration in TN, while TR showed a slight increase in anthocyanin when exposed to water stress and slight decrease as reaction to light- and heat stress. Combination of both three stresses yielded a strong increase in TN but left TR unaffected.

3.2 Change in Cell Structure

Cell structure is subject to seasonal effects as well as environmental impact and can change significantly with leaf age [26, 11]. During senescence, the reflectance decreases, which could be attributed to cell wall breakdown and the resulting changes in optical properties [26]. Plant water status has been successfully correlated to water absorption bands [33, 13]. However, the water absorption bands with strong impact on NIR spectra are not in the range of inexpensive silicon sensors and thus water status has also been indirectly correlated to the visible spectral range [29, 36]. Decreasing NIR reflectance in grapevine infected with root rot disease have been observed in [9] and attributed to structural change due to wilting of infected leaves. [32] investigated the spectral response of three red cultivars to the fungal infection *Plasmopara viticola*. They observed a slight increase in NIR reflectance for the susceptible cultivar 'Mueller-Thurgau', but a decrease in the resistant cultivars 'Regent' and 'Solaris', indicating that no general rule for structural change of leaves can be derived from specific pathogens.

4 Discussion

4.1 Spectral Imaging for Stress Identification in Viticulture

As mentioned above, stress triggers biological processes that, in turn alter the optical properties of plants. Spectral imaging has thus been successfully implemented for numerous applications for stress monitoring in viticulture.

In-field monitoring of Grapevine Leafroll disease GLRaV-1 and -3 has been demonstrated by [4] using a hyperspectral camera mounted inside a grape harvesting machine, illuminated by halogen lamps. the obtained data was analyzed by Linear Discriminant analysis (LDA), Partial Least Squares (PLS), Multilayer Perceptron (MLP) and Radial-Basis Function Network with Relevance (rRBF), achieving classification accuracies of 67-94% in the NIR- and 74-96% in the SWIR spectral range. The same algorithms have been tested in [5] to identify grapevine yellows on whole shoot hyperspectral images taken in the lab. Here, classification accuracies between 89 and 97% could be achieved. Spectral features obtained by a point spectrometer were combined with textural features extracted from RGB images in [39] to monitor Grapevine yellows and Esca. The effect of nutrient deficiency was observed by [14]. They analyzed single leaves collected in the greenhouse in a laboratory hyperspectral imaging (HSI) setup and evaluated the extracted mean leaf spectra using a binary Support Vector Machine (SVM) classifier. The suitability of HSI for water stress monitoring has been demonstrated by [29]. They used a hyperspectral camera in the visible range mounted on a tripod in the field under sunlight illumination to identify stressed vines. Making use of Random Forest and Extreme Gradient Boosting, they could correctly identify between 77 and 83% of stressed plants.

For large-scale applications, high throughput is required. Thus, the use of unmanned aerial vehicles (UAV) has become increasingly popular and has already been used in several studies. A UAV-mounted multispectral camera was utilized by [1] to map *Flavescence dorée* and Grapevine Trunk Disease. Grapevine water status was monitored by [2] using a multisensor platform carried by a UAV, including multispectral and thermal imaging systems. Airborne RGB-multispectral- and hyperspectral imaging was employed by [42] to monitor the insect pest grape phylloxera. In all these studies, sunlight provided the necessary illumination which subject to variation.

As processing hyperspectral datacubes requires substantial computational resources, the analysis is oftentimes reduced to indices and ratios of only a limited number of wavebands [1, 2, 42]. As monitoring water stress as in [2] is clearly correlated to water absorption, the use of indices for water stress monitoring is widely used. Considering biotic stresses, the change in plant status can not as clearly be attributed to the presence or absence of single substances.

Although [39, 1] could achieve satisfying classification accuracy for biotic stresses using indices from the literature, custom chemometric models based on the full spectrum could possibly achieve higher accuracy by taking into account the individual changes in pigment concentration and leaf structure. Also, generalization could be difficult since the presented studies could only observe a limited number of cultivars. As described in Section 3, cultivars can substantially differ in their reaction to stress. Thus, more research is needed to find a suitable spectral configuration for specialized cameras that can record the relevant change in pigments, cellular structure and abiotic factors, but is at the same time cost-effective and computationally efficient by limiting the number of channels.

4.2 Future Prospects

Most studies introduced above have only focused on individual cultivars. As described in Section 3.1, different cultivars show substantially divergent behavior, varying tolerance to stress influence and exhibition of symptoms. To further promote the use of spectral imaging, further work with a focus on generalization to varying environmental and biological conditions is needed. It is therefore not sufficient to extract relevant spectral bands for a single cultivar only, but also those shared by a wide range of cultivars growing under varying conditions and ideally in different climates. However, data collection of such large datasets under stable measurement conditions would be time-consuming and financially demanding. One possible solution could be the exploitation of multispectral Very High Resolution satellite imagery. Although limited in spatial and spectral resolution and flexibility, this would allow mapping of virtually any vineyard on a regular basis with comparable equipment for extended time spans and several seasons. To date, few studies have focused on the seasonal fluctuations in plant appearance, which should also be considered for further generalization.

Another common problem is given by heavily imbalanced classes. Typically, significantly more data is available for healthy reference plants than for those with known and properly defined stress or combination of stresses. Balancing techniques as suggested in e.g. [7] should be used more intensively to counteract imbalance and consequently produce more reliable models. As was evident

from the results of [10], the combination of different stresses should also be investigated in detail. In this study, only the combination of abiotic stresses was considered. Thus, further research regarding the interplay different stresses is needed for generalization.

As described above, several stresses can result in similar symptoms and similar pigment- and structural changes. Since pigment change represents an easily detectable and obvious symptom to the human observer, more general understanding on the mechanisms triggering pigment change is required across cultivars. Spectral unmixing methods could be employed for reliable pigment concentration estimation that could then be attributed to specific stresses. Some stress conditions yield characteristic patterns on leaves, e.g. the prominent stripes resulting from Esca. Thus, important information is lost if only the average spectrum of the leaf is considered. therefore, further evaluation of spectral and spatial features combined is needed, which could improve differentiation between different stresses. Additionally, combining spectral imaging with other sensing techniques such as thermography and chlorophyll fluorescence could lead to increased accuracy in diagnosing stress symptoms. This approach has been tested for disease monitoring in wheat already [30]. In grapevine, combination of multispectral and thermal imaging has been tested to monitor water status [2], but could also potentially enhance disease monitoring.

5 Conclusion

Spectral imaging is a promising technology for large-scale stress monitoring in viticulture. When exposed to stress, the pigment concentration of plants is changed, resulting in a change of coloration that is obvious on leaves. Genotypes react differently upon stress exposure, therefore spectral changes due to variation in pigment concentration cannot be generally described but must be determined for each individual cultivar. Thus, changes in pigment concentration may not be a sufficient general indicator, especially for abiotic stress. Additionally, different stress factors can influence each other and yield changes in pigment concentration that substantially differ from individual factors. Only considering spectra in the visible spectral range that is dominated by pigment absorption may

not provide sufficient evidence to differentiate between stress factors but can only give a general impression of the overall plant status and detect anomalous plants. Additional features are necessary for more selective diagnosis. These features could be represented by evaluating characteristic patterns of symptoms, e.g. discoloration in the shape of spots, stripes or whole leaf segments, which can only be obtained by imaging- but not point spectrometers. Consequently, evaluation of single pixels can be misleading since these changes oftentimes do not take place uniformly across the leaf and plant, but the symptom pattern gives more detailed insight into the type of stress. Combining spatial and spectral features, multi- and hyperspectral imaging have proven to be capable tools for stress monitoring. Further morphological information like leaf rolling or necrotic segments could provide additional evidence and could also be detected by imaging systems. Extending the spectral range under consideration to the NIR- and SWIR-spectral range can provide further information about leaf structure and water status, which could be important parameters for stress differentiation.

References

[1] Johanna Albetis et al. “On the Potentiality of UAV Multispectral Imagery to Detect Flavescence dorée and Grapevine Trunk Diseases”. In: *Remote Sensing* 11.1 (2019). DOI: 10.3390/rs11010023.

[2] Javier Baluja et al. “Assessment of vineyard water status variability by thermal and multispectral imagery using an unmanned aerial vehicle (UAV)”. In: *Irrigation Science* 30.6 (Nov. 1, 2012), pp. 511–522. DOI: 10.1007/s00271-012-0382-9.

[3] Glenn E. Bartley and Pablo A. Scolnik. “Plant Carotenoids: Pigments for Photoprotection, Visual Attraction, and Human Health”. In: *The Plant Cell* 7.7 (1995), pp. 1027–1038. DOI: 10.1105/tpc.7.7.1027.

[4] Nele Bendel et al. “Detection of Grapevine Leafroll-Associated Virus 1 and 3 in White and Red Grapevine Cultivars Using Hyperspectral Imaging”. In: *Remote Sensing* 12.10 (2020). DOI: 10.3390/rs12101693.

[5] Nele Bendel et al. “Detection of Two Different Grapevine Yellows in Vitis vinifera Using Hyperspectral Imaging”. In: *Remote Sensing* 12.24 (Dec. 18, 2020). Number: 24, p. 4151. DOI: 10.3390/rs12244151.

[6] N K Boardman. “Comparative Photosynthesis of Sun and Shade Plants”. In: *Annual Review of Plant Physiology* 28.1 (1977), pp. 355–377. DOI: 10.1146/annurev.pp.28.060177.002035.

[7] Mateusz Buda, Atsuto Maki, and Maciej A. Mazurowski. “A systematic study of the class imbalance problem in convolutional neural networks”. In: *Neural Networks* 106 (Oct. 2018), pp. 249–259. DOI: 10.1016/j.neunet.2018.07.011.

[8] C. Buschmann and E. Nagel. “In vivo spectroscopy and internal optics of leaves as basis for remote sensing of vegetation”. In: *International Journal of Remote Sensing* 14.4 (1993), pp. 711–722. DOI: 10.1080/01431169308904370.

[9] Federico Calamita et al. “Early Identification of Root Rot Disease by Using Hyperspectral Reflectance: The Case of Pathosystem Grapevine/Armillaria”. In: *Remote Sensing* 13.13 (2021). DOI: 10.3390/rs13132436.

[10] L. C. Carvalho et al. “Differential physiological response of the grapevine varieties Touriga Nacional and Trincadeira to combined heat, drought and light stresses”. In: *Plant Biology* 18.S1 (2016), pp. 101–111. DOI: 10.1111/plb.12410.

[11] Cecilia Chavana-Bryant et al. “Leaf aging of Amazonian canopy trees as revealed by spectral and physiochemical measurements”. In: *New Phytologist* 214.3 (2017), pp. 1049–1063. DOI: 10.1111/nph.13853.

[12] Elisa Costa et al. “Anthocyanin profile and antioxidant activity from 24 grape varieties cultivated in two Portuguese wine regions”. In: *OENO One* 48.1 (Jan. 31, 2014), pp. 51–62. DOI: 10.20870/oeno-one.2014.48.1.1661.

[13] R. De Bei et al. “Non-destructive measurement of grapevine water potential using near infrared spectroscopy”. In: *Australian Journal of Grape and Wine Research* 17.1 (2011), pp. 62–71. DOI: 10.1111/j.1755-0238.2010.00117.x.

[14] Sourabhi Debnath et al. “Identifying Individual Nutrient Deficiencies of Grapevine Leaves Using Hyperspectral Imaging”. In: *Remote Sensing* 13.16 (2021). DOI: 10.3390/rs13163317.

[15] Barbara Demmig-Adams, Adam M. Gilmore, and William W. Adams Iii. “In vivo functions of carotenoids in higher plants”. In: *The FASEB Journal* 10.4 (1996), pp. 403–412. DOI: 10.1096/fasebj.10.4.8647339.

[16] Samuel Eichhorn Bilodeau et al. “An Update on Plant Photobiology and Implications for Cannabis Production”. In: *Frontiers in Plant Science* 10 (2019), p. 296. DOI: 10.3389/fpls.2019.00296.

[17] Maria Lorena Falcone Ferreyra, Sebastián Rius, and Paula Casati. “Flavonoids: biosynthesis, biological functions, and biotechnological applications”. In: *Frontiers in Plant Science* 3 (2012), p. 222. DOI: 10.3389/fpls.2012.00222.

[18] Rigas Giovos et al. “Remote Sensing Vegetation Indices in Viticulture: A Critical Review”. In: *Agriculture* 11.5 (2021). DOI: 10.3390/agriculture11050457.

[19] Mahmood R. Golzarian et al. “Accurate inference of shoot biomass from high-throughput images of cereal plants”. In: *Plant Methods* 7.1 (Feb. 1, 2011), p. 2. DOI: 10.1186/1746-4811-7-2.

[20] Lois Grant, C.S.T. Daughtry, and V.C. Vanderbilt. “Polarized and specular reflectance variation with leaf surface features”. In: *Physiologia Plantarum* 88.1 (1993), pp. 1–9. DOI: 10.1111/j.1399-3054.1993.tb01753.x.

[21] Linga R. Gutha et al. “Modulation of flavonoid biosynthetic pathway genes and anthocyanins due to virus infection in grapevine (Vitis viniferaL.) leaves”. In: *BMC Plant Biology* 10.1 (Aug. 23, 2010), p. 187. ISSN: 1471-2229. DOI: 10.1186/1471-2229-10-187.

[22] Michel Havaux, Florence Tardy, and Yves Lemoine. “Photosynthetic light-harvesting function of carotenoids in higher-plant leaves exposed to high light irradiances”. In: *Planta* 205.2 (May 1, 1998), pp. 242–250. ISSN: 1432-2048. DOI: 10.1007/s004250050317.

[23] Luke Hendrickson et al. “Processes contributing to photoprotection of grapevine leaves illuminated at low temperature”. In: *Physiologia Plantarum* 121.2 (2004), pp. 272–281. ISSN: 1399-3054. DOI: 10.1111/j.0031-9317.2004.0324.x.

[24] D. N. H. Horler, M. Dockray, and J. Barber. “The red edge of plant leaf reflectance”. In: *International Journal of Remote Sensing* 4.2 (1983), pp. 273–288. DOI: 10.1080/01431168308948546.

[25] Richard E. Kendrick, ed. *Photomorphogenesis in plants*. 2. ed. Dordrecht [u.a.]: Kluwer Academic Publ., 1994. ISBN: 0792325508.

[26] Edward B. Knipling. “Physical and physiological basis for the reflectance of visible and near-infrared radiation from vegetation”. In: *Remote Sensing of Environment* 1.3 (June 1, 1970), pp. 155–159. DOI: 10.1016/S0034-4257(70)80021-9.

[27] Ole Mathis Opstad Kruse et al. “Pixel classification methods for identifying and quantifying leaf surface injury from digital images”. In: *Computers and Electronics in Agriculture* 108 (2014), pp. 155–165. DOI: 10.1016/j.compag.2014.07.010.

[28] Gaël Lebon et al. “Photosynthesis of the grapevine (Vitis vinifera) inflorescence”. In: *Tree Physiology* 25.5 (May 1, 2005), pp. 633–639. ISSN: 0829-318X, 1758-4469. DOI: 10.1093/treephys/25.5.633.

[29] Kyle Loggenberg et al. “Modelling Water Stress in a Shiraz Vineyard Using Hyperspectral Imaging and Machine Learning”. In: *Remote Sensing* 10.2 (2018). DOI: 10.3390/rs10020202.

[30] Anne-Katrin Mahlein et al. “Comparison and Combination of Thermal, Fluorescence, and Hyperspectral Imaging for Monitoring Fusarium Head Blight of Wheat on Spikelet Scale”. In: *Sensors* 19.10 (2019). DOI: 10.3390/s19102281.

[31] Paolo Margaria et al. “Metabolic and transcript analysis of the flavonoid pathway in diseased and recovered Nebbiolo and Barbera grapevines (Vitis vinifera L.) following infection by Flavescence dorée phytoplasma”. In: *Plant, Cell & Environment* 37.9 (2014), pp. 2183–2200. DOI: 10.1111/pce.12332.

[32] Erich-Christian Oerke, Katja Herzog, and Reinhard Toepfer. " Hyperspectral phenotyping of the reaction of grapevine genotypes to Plasmopara viticola". In: *Journal of Experimental Botany* 67.18 (Aug. 2016), pp. 5529–5543. DOI: 10.1093/jxb/erw318.

[33] J. Peñuelas et al. "The reflectance at the 950970 nm region as an indicator of plant water status". In: *International Journal of Remote Sensing* 14.10 (1993), pp. 1887–1905. DOI: 10.1080/01431169308954010.

[34] A.-N. Petit et al. "Alteration of Photosynthesis in Grapevines Affected by Esca". In: *Phytopathologyő* 96.10 (2006), pp. 1060–1066. DOI: 10.1094/PHYTO-96-1060.

[35] Elisa Petrussa et al. "Plant FlavonoidsBiosynthesis, Transport and Involvement in Stress Responses". In: *International Journal of Molecular Sciences* 14.7 (2013), pp. 14950–14973. DOI: 10.3390/ijms140714950.

[36] Isabel Pôças et al. "Predicting Grapevine Water Status Based on Hyperspectral Reflectance Vegetation Indices". In: *Remote Sensing* 7.12 (2015), pp. 16460–16479. ISSN: 2072-4292. DOI: 10.3390/rs71215835.

[37] John Wilson Rouse et al. "Monitoring vegetation systems in the Great Plains with ERTS". In: *NASA special publication* 351.1974 (1974), p. 309.

[38] Denis Rusjan et al. "Biochemical response of grapevine variety Chardonnay (Vitis vinifera L.) to infection with grapevine yellows (Bois noir)". In: *European Journal of Plant Pathology* 134.2 (Oct. 1, 2012), pp. 231–237. ISSN: 1573-8469. DOI: 10.1007/s10658-012-9988-2.

[39] Hania Al-Saddik et al. "Using Image Texture and Spectral Reflectance Analysis to Detect Yellowness and Esca in Grapevines at Leaf-Level". In: *Remote Sensing* 10.4 (Apr. 18, 2018), p. 618. DOI: 10.3390/rs10040618.

[40] M.G. Sajilata, R.S. Singhal, and M.Y. Kamat. "The Carotenoid Pigment ZeaxanthinA Review". In: *Comprehensive Reviews in Food Science and Food Safety* 7.1 (2008), pp. 29–49. DOI: 10.1111/j.1541-4337.2007.00028.x.

[41] Ryo Sugiura et al. “Field phenotyping system for the assessment of potato late blight resistance using RGB imagery from an unmanned aerial vehicle”. In: *Biosystems Engineering* 148 (2016), pp. 1–10. ISSN: 1537-5110. DOI: 10.1016/j.biosystemseng.2016.04.010.

[42] Fernando Vanegas et al. “Multi and hyperspectral UAV remote sensing: Grapevine phylloxera detection in vineyards”. In: *2018 IEEE Aerospace Conference*. 2018, pp. 1–9. DOI: 10.1109/AERO.2018.8396450.

[43] Thomas C. Vogelmann. “Plant Tissue Optics”. In: *Annual Review of Plant Physiology and Plant Molecular Biology* 44.1 (1993), pp. 231–251. DOI: 10.1146/annurev.pp.44.060193.001311.

[44] Amanda R. Walker et al. “White grapes arose through the mutation of two similar and adjacent regulatory genes”. In: *The Plant Journal* 49.5 (2007), pp. 772–785. ISSN: 1365-313X. DOI: 10.1111/j.1365-313X.2006.02997.x.

[45] Joseph T. Woolley. “Reflectance and Transmittance of Light by Leaves”. In: *Plant Physiology* 47.5 (May 1971), pp. 656–662. DOI: 10.1104/pp.47.5.656.

[46] Bo Zhang et al. “Copigmentation of malvidin-3-O-glucoside with five hydroxybenzoic acids in red wine model solutions: Experimental and theoretical investigations”. In: *Food Chemistry* 170 (2015), pp. 226–233. DOI: 10.1016/j.foodchem.2014.08.026.

A Transformer-based Multi-task Model for Attribute-based Person Retrieval

Andreas Specker

Vision and Fusion Laboratory
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
andreas.specker@kit.edu

Abstract

Person retrieval is a crucial task in video surveillance. While searching for persons-of-interest based on so-called query images gains much interest in the research community, attribute-based approaches are rarely studied. Attribute-based person retrieval takes a person's semantic attributes as input and provides a ranked list of search results that match the description. Typically, such approaches either build on a pedestrian attribute recognition approach or learn a joint feature space between attribute descriptions and image data. In this work, both approaches are combined in a multi-task model to benefit from the advantages of both procedures. Moreover, transformer modules are incorporated to increase performance further. Experimental evaluation proves the effectiveness of the approach and shows that the proposed architecture outperforms the baselines significantly.

1 Introduction

The task of person retrieval aims at finding persons in a large amount of image or video data. It is crucial for effective video surveillance since automatic

person retrieval assists law enforcement agencies in gathering evidence about criminals. Moreover, person retrieval techniques serve as the core component in multi-camera tracking frameworks [20, 13, 25].

Typically, person retrieval algorithms use so-called query images showing the person-of-interest [2, 30, 19] to start a search. However, such images are rarely available in real-world applications since crimes may happen in blind spots of the surveillance cameras. In such cases, attribute descriptions of the criminal gathered from eyewitnesses may be used as a query to start the search. In general, three different procedures are described in the literature. Some approaches directly use natural language queries. These approaches suffer from ambiguity issues of natural language and require complex language processing components. Moreover, merging multiple descriptions is hardly possible. Second, methods learn shared feature spaces between images and textual attribute descriptions. This procedure leads to promising results but discards the semantics of attributes by embedding them into an abstract feature space. Third, pedestrian attribute recognition (PAR) can be leveraged to identify the attributes depicted in the gallery images and match them with the query attributes during retrieval. By that, semantics is preserved, and thus retrieval results are "explainable". In addition, the search for subsets of attributes is enabled without additional computational steps.

This work studies a multi-task model to benefit from both the semantic nature of PAR approaches and the improved performance of shared feature space methods. A model is developed that simultaneously predicts the semantic attributes for images and aligns attributes and corresponding image embeddings. Since recent works indicate that transformer-based models [12, 7] are able to outperform CNN-based equivalents, the model incorporates transformer modules to further enhance performance. During inference, the model benefits from the built-in ensemble since the outputs of the PAR classifier and the joint embeddings are used in combination.

2 Related work

The straightforward approach to attribute-based person retrieval is recognizing the semantic attributes of person images and then comparing the predicted attributes with the query attribute description [27, 21, 15, 22, 24, 23]. This approach provides semantics but is difficult due to the challenging task of recognizing fine-grained, local attributes in low-resolution surveillance imagery.

Further methods align attribute descriptions and image embeddings in a shared cross-modal feature space. In doing so, the large modality gap between attributes and images has to be bridged. Approaches from the literature solve this by using high-dimensional hierarchical embeddings and an additional matching network [5]. Further works aim to match person attributes and images in a joint feature space [31, 1]. Adversarial training is applied to align the different modalities. Jeong et al. argue that this procedure is often unstable and challenging due to the min-max optimization procedure. Thus, they propose an approach that does not employ adversarial training but introduces a modality alignment loss function and a semantic regularization loss to leverage the relations between different attribute combinations explicitly [10].

This work follows a combined approach consisting of an attribute classifier and a joint feature learning method similar to [10].

3 Methods

The structure of the developed multi-task model is depicted in Figure 2.1. It consists of four main parts. The first is the backbone CNN at the bottom, which extracts feature maps from images. Transformer modules further process the outputs. Attribute features and cross-modal embedding features are computed based on the input patches. While the attribute features serve as the input for the PAR classifier, cross-modal image embeddings are aligned to the attribute embeddings produced by the attribute encoder. The fourth part of the model generates the embeddings based on the attribute descriptions that correspond to the input image. In the following, each of the parts and the loss functions are described in detail.

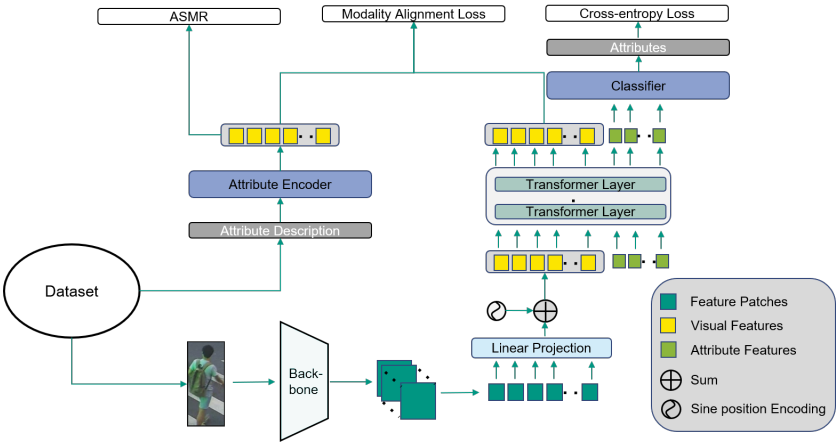


Figure 2.1: Model overview: Structure of the developed multi-task model for attribute-based person retrieval.

Backbone. The backbone model extracts feature maps from the input images. The features maps are then forwarded to the transformer part. Besides the commonly-used CNN backbones such as ResNet-50 [8], transformer-based backbone models gained increasing importance in vision tasks [7, 12]. Transformer backbones may extract better low-level features for small-scale attributes due to their attentive nature and thus improved localization. As a result, experiments in this work are conducted with both types of backbone models. Specifically, the ResNet-50 [8] CNN and the PVTv2-b2 [29] are applied. The latter extracts feature maps more efficiently, although consisting of a similar number of parameters.

Transformer-based Image Encoder. The transformer-based module is incorporated to improve the localization of the different attributes. Visual input tokens are directly extracted from the backbone feature maps. In contrast to the original transformer [28] that works with 1D token sequences as input, 2D visual tokens are leveraged as proposed by Dosovitskiy et al. [6]. For this, feature maps of size $x \in \mathbb{R}^{H \times W \times C}$ are uniformly split into $\frac{H}{P_h} \times \frac{W}{P_w}$ patches. Each patch p

Attribute	Value
# Transformer blocks M	3
# Attention heads	12
Transformer feature dimension d	382
Patch size ($P_w \times P_h$)	2×2

Table 3.1: Transformer architecture: Key facts about the transformer part of the developed model.

of size $P_h \times P_w$. Each patch p is subsequently flattened to obtain a 1D vector with $P_h \cdot P_w \cdot C$ elements. Last, visual tokens v are obtained by projecting flattened patches into a d-dimensional embedding space using a linear function $f : p \longrightarrow v \in \mathbb{R}^d$. Position embeddings [28] pe_i are added to each visual token to encode that each visual token represents a specific area of the input image,

Since the aim is to retrieve cross-modal attribute-image embeddings and features for attribute classification, attribute tokens are additionally incorporated analogous to recent works [18, 9]. For each of the N binary attributes, a learnable, d-dimensional attribute token is concatenated to the visual token. In the following, we refer to these attribute tokens as `[attribute]`. The sequence $T = \{\texttt{[visual]}, \texttt{[attribute]}\}$ then serves as input for the transformer part. It consists of M stacked transformer modules, each of which contains a Multi-head Self-attention bock followed by a Multi-layer Perceptron with layernorm before every block. The outputs are the states of the visual tokens and the N attribute tokens. While the classifier further processes the attribute tokens, global average pooling (GAP) and one fully-connected layer are applied to the visual tokens to obtain the cross-modal 128-dimensional image embedding for cross-modal matching. Details about the parameterization of the transformer blocks are given in Table 3.1.

Pedestrian Attribute Recognition. The states of the attribute tokens `[attribute]` are used as input for the attribute classifier. Fully-connected classification layers generate confidence scores based on the token features of the respective attributes. The result is an N-dimensional vector containing the attributes' confidence scores.

Structure	Size
FC_1	$N \times 512$ ReLU
FC_2	512×128 ReLU
FC_3	128×128

Table 3.2: Structure of the attribute encoder network. It consists of three fully-connected layers and was taken from [10].

Attribute Encoder. The attribute encoder is taken from the work of Jeong et al. [10]. The configuration is provided in Table 3.2. Input is the N-dimensional binarized attribute label vector. Binarization is done by concatenating single elements for binary attributes and one-hot vectors for multi-class attributes. After three fully-connected layers with ReLU activation functions in between, the final cross-modal features with 128 dimensions are obtained.

Loss Functions. Three different loss functions are employed to train the multi-task model: one for the PAR task and two for the modality alignment task.

Most works [14, 16, 17, 26] on PAR consider the task a multi-label classification problem. Analogous to that, multiple binary classifiers with Sigmoid activation functions are used in this work. Therefore, the binary cross-entropy loss function fits the aim of the optimization:

$$L_{PAR} = \frac{1}{K} \sum_{i=1}^{K} \sum_{j=1}^{N} [y_{i,j} \log(p_{i,j}) + (1 - y_{i,j}) \log(1 - p_{i,j})] * w_j \qquad (3.1)$$

While K denotes the number of training images in the dataset, $y \in \{0, 1\}^N$ stands for the ground truth attribute vector and p for the predicted attribute confidences after the Sigmoid activation layer. Moreover, w_j represents an attribute-specific weighting factor in tackling the problem of imbalanced attribute distributions [14].

Regarding modality alignment (MA), the loss function (L_{MA}) proposed by Jeong et al. [10] that was inspired by the ArcFace [3] loss is utilized. In addition, the Adaptive Semantic Margin Regularizer (ASMR) [10] is applied to balance the distances between attribute embeddings in the joint feature space based on the semantic similarity measured by the Hamming distance between binary attribute vectors. This loss function is termed L_{ASMR} in the following.

The resulting loss function is formulated as follows:

$$L_{total} = L_{PAR} + \lambda_{MA} * L_{MA} + \lambda_{ASMR} * L_{ASMR} \tag{3.2}$$

Further Improvements. Two additional enhancements are investigated in this work and described in the following.

The original calculation of the MA and ASMR losses only considers the attribute sets included in the training data. However, the limited number of training images only constitutes a small excerpt of reality. Since further plausible attribute combinations could be easily generated, further experiments use additional reasonable attribute combinations (AAC) to compute these losses. AAC may improve the generalization concerning persons with previously unseen attribute sets.

The second enhancement relates to the ensemble-like nature of the approach. To support the learning of complementary strengths, the influence of mutual weighting of the two tasks is examined. For example, samples with high loss values in one task are given a higher weight in calculating the loss of the other task. The feedback mechanism should strengthen the complementary aspect of the two tasks by focusing on the weaknesses of the other.

Retrieval. Separate distance matrices are computed based on the PAR outputs and the cross-modal features to perform the retrieval. The Euclidean distance between binary attribute query vectors and confidence scores from the attribute classifier is calculated for PAR. In contrast, query vectors are embedded using the attribute encoder to obtain the distances in the joint feature space. Subsequently, the Cosine distance to the image embeddings of gallery samples is calculated. Last, both distance matrices are normalized by their highest values to achieve scores in the range of 0 to 1, and the weighted sums serve as the final retrieval

Dataset	PETA
# Binary attributes N	35
# Train images K	12,140
# Test images	1,181

Table 4.1: Statistics of the PETA [4] dataset.

distances. The weighting factor λ_{Ret} is applied to the distances from the PAR-based retrieval.

4 Evaluation

In this section, the developed architecture is evaluated to demonstrate its effectiveness and gain insights into the proposed extensions.

Datasets. Evaluation is performed on the PETA [4] dataset. The number of binary attributes N for the dataset is 35. Please note that this follows previous works, which typically use only the 35 most frequent attributes. Table 4.1 provides an overview of the dataset.

Parameters & training setup. Concerning the hyper-parameters of the MA and ASMR loss functions, the values proposed in the original work are applied [10]. PAR baselines are trained as described in [11]. The pre-trained weights of these PAR baselines are used to initialize the backbone. The model is trained using the SGD optimizer with values 0.9 and $5\mathrm{e}{-4}$ for up to 20 epochs. Batches include 16 training samples, and the initial learning rates are set to $1\mathrm{e}{-2}$ for the attribute encoder and $1\mathrm{e}{-3}$ for the remaining model parts. All learning rates are decayed by 0.1 after 10 epochs. Regarding loss weights, λ_{ASMR} values are chosen as proposed in [10], while λ_{MA} is set to 1, i.e., both tasks contribute equally to the total loss. During retrieval, $\lambda_{Ret} = 0.2$ is used to weight the contribution of the PAR task.

Evaluation Metrics. The experiments in this work are evaluated using the Mean Average Precision (mAP) and the Rank 1 score from the Cumulative

Method	mAP	Rank1
PAR Baseline (ResNet-50)	20.0	20.7
PAR Baseline (PVTv2-b2)	21.0	21.5
MA+ASMR (ResNet-50)	20.2	20.0
MA+ASMR (PVTv2-b2)	21.1	20.7
Multi-task Transformer (ResNet-50)	22.7	**22.7**
Multi-task Transformer (PVTv2-b2)	**22.9**	**22.7**

Table 4.2: Comparison of single-task baselines with the introduced multi-task model. Bold numbers denote the best results.

Matching Characteristics (CMC) curve. While the mAP measures the quality of the entire rankings, the Rank 1 accuracy represents the portion of queries in the test set that show a match at the first position in the rank list.

Method	mAP	Rank1
Multi-task Transformer	22.7	22.7
+ AAC	23.1	22.7
+ Feedback MA	23.5	22.6
+ Feedback PAR	**23.7**	23.0
+ Feedback MA&PAR	23.6	23.2
+ Feedback MA&PAR and AAC	**23.7**	**23.3**

Table 4.3: Evaluation of proposed extensions. Bold numbers denote the best results.

Discussion. Table 4.1 contains a comparison of two baseline methods with the proposed approach. Single-task models for PAR [11] and learning a joint feature space [10] serve as baselines. One can observe that the proposed multi-task transformer outperforms the baseline with respect to both evaluation metrics. Using the ResNet-50 as the backbone and the PAR model as a comparison

method, the mAP increases by +2.7% points and the Rank 1 accuracy by +2% points, respectively. Concerning the backbone models, the results indicate no significant difference. While the baseline methods benefit from a transformer-based backbone model, the performance gap is negligible for the proposed approach. This finding indicates that the use of a transformer head is sufficient to achieve good localization of small-scale attributes.

Next, the use of additional attribute categories during training and the cross-task feedback is evaluated in Table 4.3. All the proposed adoptions lead to improvements. In the case of AAC and MA loss feedback, this is limited to the mAP. In contrast, weighting the PAR loss based on the MA loss improves both metrics. Combining both types of feedback further enhances the Rank-1 accuracy and jointly using the additional attribute categories and the feedback leads to the best results.

5 Conclusion and Future Work

In this work, the idea of a transformer-based multi-task model for cross-modal attribute-based person retrieval was presented. Instead of relying on either PAR or learning of joint feature space, both approaches are combined to obtain the flexibility and the semantics of PAR methods while benefiting from the strong retrieval performances of the second procedure. Experimental results indicate that the multi-task model outperforms the single-task baselines.

However, future improvements may include a detailed evaluation of the model architecture and a more sophisticated approach for combining the results of both tasks during retrieval. In addition, it could be observed that optimal training hyper-parameters differ for both tasks. Further investigations are necessary to improve the training procedure and thus the results.

References

[1] Yu-Tong Cao, Jingya Wang, and Dacheng Tao. "Symbiotic Adversarial Learning for Attribute-based Person Search". In: *Proc. European Conference on Computer Vision (ECCV)*. 2020.

[2] Guangyi Chen et al. "Self-Critical Attention Learning for Person Re-Identification". In: *Proc. IEEE International Conference on Computer Vision (ICCV)*. Oct. 2019.

[3] Jiankang Deng et al. "Arcface: Additive angular margin loss for deep face recognition". In: *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019.

[4] Yubin Deng et al. "Pedestrian attribute recognition at far distance". In: *Proc. ACM Multimedia Conference (ACMMM)*. 2014.

[5] Qi Dong, Shaogang Gong, and Xiatian Zhu. "Person search by text attribute query as zero-shot learning". In: *Proc. IEEE International Conference on Computer Vision (ICCV)*. 2019.

[6] Alexey Dosovitskiy et al. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. 2021. arXiv: 2010.11929 [cs.CV].

[7] Kai Han et al. *A Survey on Vision Transformer*. 2021. arXiv: 2012.12556 [cs.CV].

[8] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 770–778.

[9] Shuting He et al. "TransReID: Transformer-Based Object Re-Identification". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. Oct. 2021, pp. 15013–15022.

[10] Boseung Jeong, Jicheol Park, and Suha Kwak. "ASMR: Learning Attribute-Based Person Search with Adaptive Semantic Margin Regularizer". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, pp. 12016–12025.

[11] Jian Jia et al. “Rethinking of Pedestrian Attribute Recognition: Realistic Datasets with Efficient Method”. In: *arXiv preprint arXiv:2005.11909* (2020).

[12] Salman Khan et al. *Transformers in Vision: A Survey*. 2021. arXiv: 2101.01169 [cs.CV].

[13] Philipp Kohl et al. “The MTA Dataset for Multi-Target Multi-Camera Pedestrian Tracking by Weighted Distance Aggregation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. June 2020.

[14] Dangwei Li, Xiaotang Chen, and Kaiqi Huang. “Multi-attribute Learning for Pedestrian Attribute Recognition in Surveillance Scenarios”. In: *ACPR*. 2015, pp. 111–115.

[15] Dangwei Li, Xiaotang Chen, and Kaiqi Huang. “Multi-attribute learning for pedestrian attribute recognition in surveillance scenarios”. In: *IAPR Asian Conference on Pattern Recognition (ACPR)*. 2015.

[16] Dangwei Li et al. “A richly annotated dataset for pedestrian attribute recognition”. In: *arXiv preprint arXiv:1603.07054* (2016).

[17] Dangwei Li et al. “A richly annotated pedestrian dataset for person retrieval in real surveillance scenarios”. In: *IEEE transactions on image processing* 28.4 (2018), pp. 1575–1590.

[18] Yanjie Li et al. “TokenPose: Learning Keypoint Tokens for Human Pose Estimation”. In: *IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021.

[19] Jicheol Park et al. “Learning Discriminative Part Features Through Attentions For Effective And Scalable Person Search”. In: *IEEE International Conference on Image Processing (ICIP)*. 2020.

[20] Ergys Ristani et al. *Performance Measures and a Data Set for Multi-Target, Multi-Camera Tracking*. 2016. arXiv: 1609.01775 [cs.CV].

[21] Walter J Scheirer et al. “Multi-attribute spaces: Calibration for attribute fusion and similarity search”. In: *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2012.

[22] Arne Schumann, Andreas Specker, and Jürgen Beyerer. “Attribute-based person retrieval and search in video sequences”. In: *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE. 2018, pp. 1–6.

[23] Andreas Specker and Jürgen Beyerer. “Improving Attribute-Based Person Retrieval By Using A Calibrated, Weighted, And Distribution-Based Distance Metric”. In: *2021 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2021, pp. 2378–2382.

[24] Andreas Specker, Arne Schumann, and Jürgen Beyerer. “An evaluation of design choices for pedestrian attribute recognition in video”. In: *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE. 2020, pp. 2331–2335.

[25] Andreas Specker et al. “An Occlusion-Aware Multi-Target Multi-Camera Tracking System”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. June 2021, pp. 4173–4182.

[26] Chufeng Tang et al. “Improving Pedestrian Attribute Recognition With Weakly-Supervised Multi-Scale Attribute-Specific Localization”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2019, pp. 4997–5006.

[27] Daniel A Vaquero et al. “Attribute-based people search in surveillance environments”. In: *Proc. Winter Conference on Applications of Computer Vision (WACV)*. 2009.

[28] Ashish Vaswani et al. *Attention Is All You Need*. 2017. arXiv: 1706.03762 [cs.CL].

[29] Wenhai Wang et al. “Pvtv2: Improved baselines with pyramid vision transformer”. In: *arXiv preprint arXiv:2106.13797* (2021).

[30] Yichao Yan et al. “Learning Context Graph for Person Search”. In: *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019.

[31] Zhou Yin et al. “Adversarial Attribute-Image Person Re-identification”. In: *Proc. International Joint Conferences on Artificial Intelligence (IJCAI)*. 2018.

Multi-Person Tracking with a Multi-Hypothesis Approach for Ambiguous Assignments

Daniel Stadler

Vision and Fusion Laboratory
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
daniel.stadler@kit.edu

Abstract

Multi-person tracking is often solved with the tracking-by-detection paradigm, in that a distance measure is calculated for each possible track-detection assignment. Then, the sum of distances of all assignments has to be minimized, for which mostly the Hungarian method is used. Wheareas it is easy to design a distance measure that can clearly indicate the correct assignments in sequences with sparse person distributions, the distances of some assignments can be very similar in crowded scenes, where multiple persons share similar spatial positions and appearances. As a consequence, wrong assignments are inescapable, harming the tracking performance. In contrast of executing all assignments simultaneously, no matter if they are clear or ambiguous, this work treats ambiguous assignments with similar distances separately following a multi-hypothesis approach, updating the hypotheses until the assignment task is clear again. To determine which assignments are considered ambiguous, a method that compares the entries in the distance matrix of track-detection assignments is introduced. No further information next to the distance matrix is needed, which makes the proposed approach applicable to any tracking-by-detection based method. Experimental results show that the separate treatment of ambiguous assignments can improve the tracking performance in crowds and thus is a promising research directory.

1 Introduction

Multi-person tracking (MPT) is the task of detecting and identifying all persons in each frame of a video and is the basis for several applications ranging from action classification to crowd behavior analysis.

Most of the MPT approaches in literature follow the *tracking-by-detection* paradigm [2, 3, 7, 12, 14, 15, 16, 17, 18], which divides the problem into two subtasks: detection and association. In each iteration, the generated detections are assigned to the tracks from the previous time step on the basis of a distance measure, whereby mostly the Hungarian method [5] is applied for minimizing the overall costs. When designing the distance measure, different target information can be leveraged. Especially position information [2, 3] and visual cues [7, 15, 16, 18] are used. Some works additionally consider human poses [15, 17] or relation information w.r.t. other targets [7, 13, 18]. Designing such sophisticated distance measures aims to achieve a high degree of distinguishability between correct and incorrect assignments. However, there will still exist some situations, in that the assignment task is ambiguous, no matter how good the designed distance measure is. This holds especially true in crowded scenes, where multiple targets share similar positions and appearances. Furthermore, inaccurate and missing detections under heavy occlusion often prevent a clear association.

Because of the aforementioned reasons, a new association strategy is proposed, which treats *ambiguous* assignments separately with a multi-hypothesis approach, while solving the *clear* assignments with the Hungarian method as usual. To find ambiguous assignments between detections and tracks (and track hypotheses), a closer look at the distance matrix is taken. More precisely, the distances of all possible track-detection assignments are compared and if they differ by less than a similarity threshold, the involved tracks and detections are termed *similar*. If additionally, the numbers of tracks and detections that lead to similar assignments are different, i.e., either tracks or detections would remain unassigned, the assignments are considered ambiguous and multiple track hypotheses are built. These are updated in consecutive frames until the assignment task is clear again. With this strategy, ambiguous association decisions as often occurring under occlusion can be postponed und thus assignment errors prevented. As the determination of ambiguous assignments builds purely upon the distance matrix,

the proposed approach can be included in any tracking-by-detection method, independent from how the distances are calculated.

2 Find Ambiguous Assignments

In each time step t of a tracking-by-detection based approach, detections $\mathcal{D}^{(t)} = \{D_1^{(t)}, D_2^{(t)}, \dots, D_N^{(t)}\}$ are matched to tracks from the previous iteration $\mathcal{T}^{(t-1)} = \{T_1^{(t-1)}, T_2^{(t-1)}, \dots, T_M^{(t-1)}\}$. For each track $T_i \in \mathcal{T}^{(t-1)}$ and detection $D_j \in \mathcal{D}^{(t)}$, a distance $d(T_i, D_j)$ is calculated and saved at the respective position (i, j) in the distance matrix $\mathbf{D} \in \mathbb{R}^{M \times N}$. Hereby, various information can be used. For example, position, appearance, or motion cues are often considered. In contrast to the standard association, which applies the Hungarian method on the full distance matrix $\mathbf{D}$, it is argued that some detections and tracks might lead to *ambiguous* assignments, that should be treated separately. After that, the remaining *clear* assignments are handled by the Hungarian method.

In the following, a subset of possible assignments including track indices $\mathcal{I} \subset \{1, \dots M\}$ and detection indices $\mathcal{J} \subset \{1, \dots N\}$ is noted as tuple of sets $A = (\mathcal{I}, \mathcal{J}) = (A[1], A[2])$. For example, the possible assignments A of tracks T_1, T_3 and detections D_2, D_4 are noted as $A = (\{1, 3\}, \{2, 4\})$. The search for ambiguous assignments starts with *similar* assignments. Assignments are termed similar, if the respective entries in the distance matrix differ by less than a similarity threshold Δ. For example, the possible assignments of track T_1 to detection D_1 and of track T_1 to D_2 are similar, if $|\mathbf{D}[1, 1] - \mathbf{D}[1, 2]| < \Delta$ holds. If multiple detections *and* tracks are similar, the distance matrix $\mathbf{D}$ has to be scanned several times, iteratively searching for similar detections and tracks. The process for finding all similar assignments w.r.t. a specific detection (D_2) is shown for a toy example distance matrix in Figure 2.1. This procedure is done for each detection leading to a tentative set of similar assignments $\tilde{\mathcal{A}}^{\text{sim}} = \{A_j\}_{j=1\dots N}$. Then, the assignments that share track or detection indices are merged leading to the final set of similar assignments $\mathcal{A}^{\text{sim}}$. For instance, $\tilde{\mathcal{A}}^{\text{sim}} = \{(\{1, 2\}, \{3\}), (\{3\}, \{3, 4\})\}$ would turn to $\mathcal{A}^{\text{sim}} = \{(\{1, 2, 3\}, \{3, 4\})\}$ applying the merging operation. After determining $\mathcal{A}^{\text{sim}} = \{A_k\}_{k=1\dots K}$, it is decided for each element, whether the similar

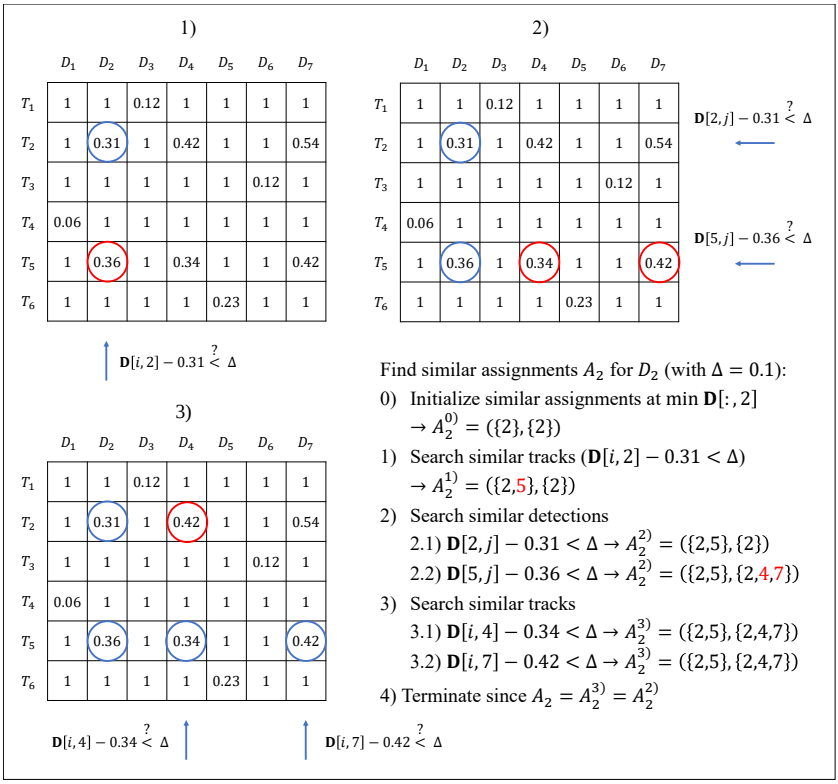


Figure 2.1: Process of finding similar assignments A_2 for detection D_2. **0)** Initialization of A_2 is done at the minimum distance of tracks w.r.t. D_2 which is $\mathbf{D}[2,2] = 0.31$ for track T_2 leading to $A_2^{0)} = (\{2\},\{2\})$. **1)** Similar tracks with lower distance difference to 0.31 than Δ are searched which yields T_5 (highlighted in red) and $A_2^{1)} = (\{2,5\},\{2\})$. **2)** Similar detections w.r.t. T_2 and T_5 are searched. D_4 and D_7 are added to the similar assignments $A_2^{2)} = (\{2,5\},\{2,4,7\})$. **3)** Again, similar tracks are searched, now for D_4 and D_7 yielding only T_2 which is already represented in $A_2^{2)}$. Thus $A_2^{3)}$ equals $A_2^{2)}$. **4)** Since A did not change in iteration 3), the algorithm stops, outputting $A_2 = A_2^{3)} = (\{2,5\},\{2,4,7\})$ as similar assignments for D_2.

assignments A_k are considered as ambiguous or not. It is argued that, if the number of tracks $n_T = |A_k[1]|$ and the number of detections $n_D = |A_k[2]|$ that

lead to similar assignments A_k is identical, the situation is not ambiguous, since each track and detection can be matched. This holds especially true for similar assignments with only one track and one detection ($n_T = n_D$). Those assignments are termed as *clear*, while similar assignments, for which the numbers of tracks and detections differ ($n_T \neq n_D$), i.e., there are missing detections or missing tracks, are termed *ambiguous*. Formally, the set of similar assignments $\mathcal{A}^{\text{sim}}$ is divided into a set of ambiguous assignments $\mathcal{A}^{\text{amb}}$ and a set of clear assignments $\mathcal{A}^{\text{clr}}$:

$$\mathcal{A}^{\text{amb}} = \{A | A \in \mathcal{A}^{\text{sim}} \wedge A[1] \neq A[2]\} \tag{2.1}$$

$$\mathcal{A}^{\text{clr}} = \{A | A \in \mathcal{A}^{\text{sim}} \wedge A[1] = A[2]\} \tag{2.2}$$

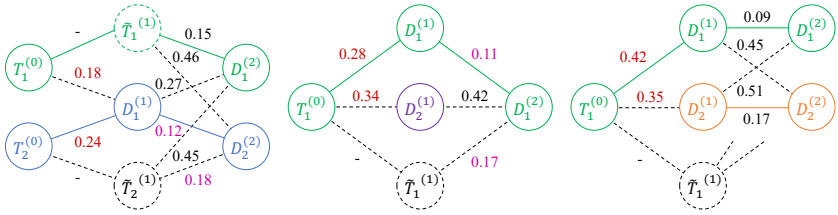
Note that $\mathcal{A}^{\text{sim}} = \mathcal{A}^{\text{amb}} \cup \mathcal{A}^{\text{clr}}$ holds. The track indices $\mathcal{I}^{\text{clr}}$ and detection indices $\mathcal{J}^{\text{clr}}$ of the clear assignments $\mathcal{A}^{\text{clr}}$ are used to generate a clear distance matrix $\mathbf{D}^{\text{clr}} = \mathbf{D}[\mathcal{I}^{\text{clr}}, \mathcal{J}^{\text{clr}}]$ on which the Hungarian method is applied. In contrast, the ambiguous assignments $\mathcal{A}^{\text{amb}}$ are treated separately with a multi-hypothesis tracking (MHT) approach that is described in the next section.

3 Solve Ambiguous Assignments with MHT

For each group of tracks and detections that lead to ambiguous assignments, multiple hypotheses are started and thus the assignment problem is postponed. In consecutive iterations, the set of hypotheses is updated until the assignment task is clear again. In the following, the procedure of building and updating the set of track hypotheses is exemplary described for an ambiguous situation with missing detection (FN). This situation is also depicted in Figure 3.1(a), however, note that the full figure has not to be understood at this point.

At time step $t = 1$, there is only one detection $D_1^{(1)}$ but there have been two tracks $T_1^{(0)}$ and $T_2^{(0)}$ in the previous iteration. Furthermore, the detection fits nearly equally well to both tracks ($0.24 - 0.18 < \Delta = 0.1$) so ambiguous assignments $A = (\{1, 2\}, \{1\}) \in \mathcal{A}^{\text{amb}}$ are present. Therefore, the two hypotheses $H_{11}^{(1)}$ and $H_{21}^{(1)}$ are built:

$$H_{11}^{(1)} = [T_1^{(0)}, D_1^{(1)}] \qquad H_{21}^{(1)} = [T_2^{(0)}, D_1^{(1)}] \tag{3.1}$$



(a) FN (b) FP (c) New detection

Figure 3.1: Illustration of the three types of ambiguous assignments ($\Delta = 0.1$) that can be handled by the proposed multi-hypothesis approach: (a) false negatives (FN), (b) false positives (FP), and (c) new detections which initialize new tracks. Detections and tracks are visualized with circles, whereby propagated tracks (without assigned detection) are indicated with dashed circles and a tilde. Hypotheses with the smallest distance that correspond to the resulting trajectories are marked in green, blue, and orange. Hypotheses with higher distances are drawn in dashed lines. Detections that are removed are depicted purple. Distances between detections and tracks are written near the edges of the graph. Distances that lead to ambiguous assignments are highlighted in red, whereas assignments with similar distances that can be clearly resolved are highlighted in pink. Note that there is no distance (-) for propagated tracks. Furthermore, time steps are specified in superscript and for situation (c), hypotheses with the propagated track are omitted in the second step for clarity.

In addition, two hypotheses $H_{10}^{(1)}$ and $H_{20}^{(1)}$ that are based on a Kalman filter prediction step (see details about the motion model in Section 4.2) are started:

$$H_{10}^{(1)} = [T_1^{(0)}, \tilde{T}_1^{(1)}] \qquad H_{20}^{(1)} = [T_2^{(0)}, \tilde{T}_2^{(1)}] \tag{3.2}$$

Note that propagated tracks are indicated with a tilde. The overall set of track hypotheses $\mathcal{H}^{(1)} = \{H_{10}^{(1)}, H_{11}^{(1)}, H_{20}^{(1)}, H_{21}^{(1)}\}$ is saved for the next time step. At $t = 2$, there are two detections $D_1^{(2)}$ and $D_2^{(2)}$ that update the set of hypotheses to $\mathcal{H}^{(2)} = \{H_{101}^{(2)}, H_{102}^{(2)}, H_{111}^{(2)}, H_{112}^{(2)}, H_{201}^{(2)}, H_{202}^{(2)}, H_{211}^{(2)}, H_{212}^{(2)}\}$ with:

$$\begin{aligned}
H_{101}^{(2)} &= [T_1^{(0)}, \tilde{T}_1^{(1)}, D_1^{(2)}] & H_{102}^{(2)} &= [T_1^{(0)}, \tilde{T}_1^{(1)}, D_2^{(2)}] \\
H_{111}^{(2)} &= [T_1^{(0)}, D_1^{(1)}, D_1^{(2)}] & H_{112}^{(2)} &= [T_1^{(0)}, D_1^{(1)}, D_2^{(2)}] \\
H_{201}^{(2)} &= [T_2^{(0)}, \tilde{T}_2^{(1)}, D_1^{(2)}] & H_{202}^{(2)} &= [T_2^{(0)}, \tilde{T}_2^{(1)}, D_2^{(2)}] \\
H_{211}^{(2)} &= [T_2^{(0)}, D_1^{(1)}, D_1^{(2)}] & H_{212}^{(2)} &= [T_2^{(0)}, D_1^{(1)}, D_2^{(2)}]
\end{aligned} \tag{3.3}$$

For each hypothesis $H \in \mathcal{H}^{(2)}$, a distance $d(H)$ has to be determined in order to find out whether the assignment problem is still ambiguous or clear again. The distance of a hypothesis is set to the distance between the last two entries of the hypothesis. For instance, the distance of $H_{111}^{(2)} = [T_1^{(0)}, D_1^{(1)}, D_1^{(2)}]$ is $d(H_{111}^{(2)}) = d(D_1^{(1)}, D_1^{(2)}) = 0.27$. With the hypothesis distances and the information about track and detection numbers n_T and n_D, respectively, within a set of hypotheses, one of the following two requirements has to be fulfilled that the assignment problem is considered clear again:

1. Track number and detection number are identical: $n_T = n_D$.
2. There are more detections than tracks ($n_D > n_T$) in each iteration *and* the number of detections is the same for two successive time steps: $n_D^{(t)} = n_D^{(t-1)}$.

The first item corresponds to cases (a) and (b) and the second item applies for case (c) in Figure 3.1. The three types of ambiguous assignments (false negatives, false positives, new detections) are discussed in the following subsections.

The attentive reader may have noticed that not all distances of the eight hypotheses in $\mathcal{H}^{(2)}$ are depicted in Figure 3.1(a). As the assignment of a detection to a track changes its motion prediction, the propagated track position differs for two tracks that would be assigned the same detection. Thus, for example, $d(H_{111}^{(2)}) \neq d(H_{211}^{(2)})$ holds, which is not considered in Figure 3.1(a) for clarity. Another side note is that the overall number of hypotheses is limited to maintain a low computational complexity when multiple time steps are involved. More precisely, the $h_{\max}$ hypotheses with the lowest distances are kept in each iteration.

3.1 Missing Detections

Missing detections (FN) appear frequently in crowded scenes, where the detector cannot recognize all targets due to occlusion. At the same time, the assignment of the available detections can be ambiguous due to inaccuracies of the detection boxes or the propagated track boxes. This is also the case for $D_1^{(1)}$ in Figure 3.1(a) which can be assigned either to $T_1^{(0)}$ ($d = 0.18$) or to $T_2^{(0)}$ ($d = 0.24$) as already

discussed. In the next iteration, however, the assignment becomes clear as the distance $d(D_1^{(1)}, D_2^{(2)}) = 0.12$ is significantly lower (in terms of $\Delta = 0.1$) than $d(D_1^{(1)}, D_1^{(2)}) = 0.27$. Therefore, the assignment task is clear again and the ambiguous situation can be resolved. Here, three things should be noted. First, with the multi-hypothesis approach, an identity switch is prevented, since detection $D_1^{(1)}$ would be erroneously assigned to $T_1^{(1)}$ in the standard association. Second, the missing detection for track T_1 is bridged by track propagation with the motion model. Third, the two hypotheses $H_{202}^{(2)}$ and $H_{212}^{(2)}$ are similar (pink) but not ambiguous, as hypotheses with propagated track boxes are not considered competing to hypotheses of the same track with a detection assigned.

3.2 Duplicate Detections

In crowded scenes, it can also happen that the detector produces duplicate detections (FP), struggling to recognize the precise boundaries of the targets. In the simplest case, two detections $D_1^{(1)}$ and $D_2^{(1)}$ fit nearly equally well to a track $T_1^{(0)}$ as in Figure 3.1(b). In the standard association, that starts new tracks with unassigned detections, an additional duplicate track would be initialized that could introduce further tracking errors. In contrast, the multi-hypothesis approach postpones the decision, which detection should be assigned to the track, until the situation is clear again. Then, the unassigned duplicate detection (purple) can be identified and removed. Note that it would also be possible that the propagated track box $\tilde{T}_1^{(1)}$ is the best match to $D_1^{(2)}$, namely if both detection boxes $D_1^{(1)}$ and $D_2^{(1)}$ are inaccurate due to occlusion. In that case, the propagated track $\tilde{T}_1^{(1)}$ would be used and both detections removed.

3.3 New Detections

In the previous subsection, a situation with $n_D > n_T$ has been treated, where the additional detections have been considered as FP. However, this is not the case, when a new target is about to show for the first time, still partly occluded by a nearby target. Fortunately, the proposed multi-hypothesis approach can identify such situations implicitly as shown in Figure 3.1(c). Whereas the situation is similar to case (b) in the first time step, two detections are again

present in the second iteration. Furthermore, the assignment task is clear at $t = 2$. Therefore, it is likely that the additional detections belong to a separate target, also because FP mostly occur only in single frames. As a consequence, the hypothesis with the smallest distance of 0.09 (green) is taken for track $T_1^{(0)}$, whereas $D_2^{(1)}$ and $D_2^{(2)}$ start a new track $T_{\text{new}}^{(2)} = [D_2^{(1)}, D_2^{(2)}]$ (orange). Note that an identity switch is prevented with the multi-hypothesis approach that postpones the assignment decision until the situation is clear again.

4 Experiments

4.1 Dataset and Evaluation Metrics

The MOT17 dataset [8] is used in the experiments, since it is one of the most popular benchmarks for evaluating multi-target tracking performance. It comprises a train and a test split with 7 videos each. Since the annotations of the test split are not publicly available, the train split is divided into two halves, enabling the evaluation of the tracker in combination with a fine-tuned detection model, which is necessary for achieving good results. More precisely, the images of the first part of each sequence are taken for fine-tuning the detector, while the tracker is evaluated on the second parts of the sequences.

The tracking performance is measured in IDF1 [10], that emphasizes on identity preservation abilities, and MOTA [1], which focuses more on detection quality. Furthermore, the components of MOTA are reported, i.e., number of false negatives (FN), false positives (FP), and identity switches (IDSW).

4.2 Implementation Details

As detection model for the tracking-by-detection based approach, a Faster R-CNN [9] with FPN [6] and ResNet-50 [4] as backbone is used. The model is first pre-trained on the CrowdHuman dataset [11] with a batch size of 16 and an initial learning rate of 0.01 for 30 epochs, which is lowered by factor 10 after epochs 24 and 27. Then, the model is fine-tuned on the first half of MOT17 with an initial learning rate of 0.001 with the same schedule. For the tracking process,

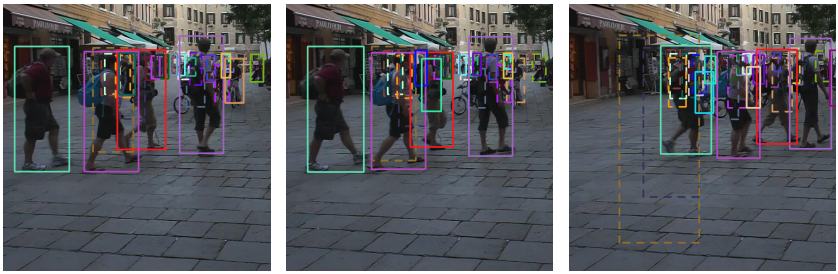
the generated detections are filtered with a minimum score threshold of 0.9 and an Intersection over Union (IoU) threshold of 0.5 is applied in the non-maximum suppression (NMS) step. The distance measure between a track T and detection D is calculated as $d(T, D) = 1 - \mathrm{IoU}(T, D)$. Both in the standard association and the proposed multi-hypothesis approach for ambiguous assignments, a maximum distance of $d_{\max} = 0.8$, which corresponds to a minimum IoU of 0.2, is enforced for matching tracks and detections. Tracks are propagated with a Kalman filter as motion model, whereby the implementation of [16] is used. Inactive tracks and track hypotheses are maintained for a maximum number of 40 iterations without assigned detection before termination or deletion. At re-activation, a linear interpolation is performed to close the gap of missed detections. For finding ambiguous assignments in the distance matrix $\mathbf{D}$, a similarity threshold $\Delta = 0.1$ is applied. The number of track hypotheses h is limited to $h_{\max} = 10$ to keep a low computational complexity of the approach.

4.3 Results

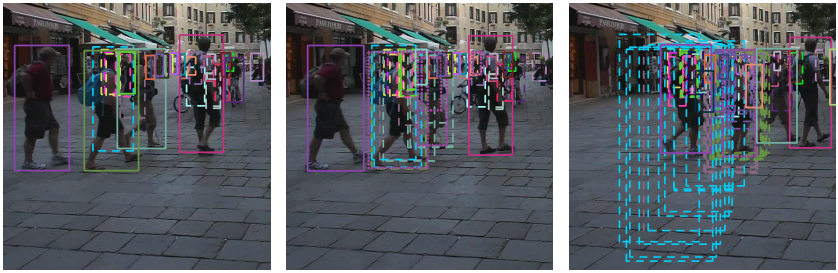
To get a feeling, in which situations the proposed multi-hypothesis approach for ambiguous assignments is superior to the standard association, in that all assignments are made with the Hungarian method, several experiments with different settings are run. The results are summarized in Table 4.1.

MH ($n_D < n_T$)	MH ($n_D > n_T$)	IDF1	MOTA	FN	FP	IDSW
✗	✗	76.1	73.2	24798	18036	600
✓	✗	**77.8**	72.6	**24264**	19593	597
✗	✓	76.9	**73.6**	25935	**16218**	**591**
✓	✓	76.8	73.1	25563	17403	615

Table 4.1: Tracking results of the proposed multi-hypothesis approach for ambiguous assignments in comparison with the standard association (first row). MH ($n_D < n_T$) corresponds to applying the multi-hypothesis approach only in situations with missing detections, while in the MH ($n_D > n_T$) variant, the approach is only applied in situations with more detections than tracks. The last row shows results, where for all ambiguous assignments, multiple track hypotheses are built.



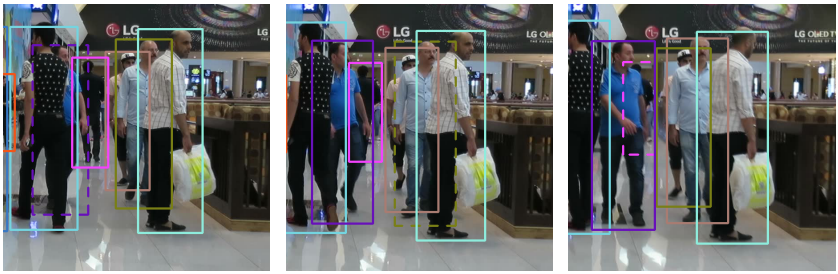
(a) Baseline



(b) Proposed approach

Figure 4.1: Failure case of the proposed multi-hypothesis approach. Active tracks are drawn in solid lines, inactive tracks or track hypotheses in dashed lines. (a) In the standard association, some incorrect inactive tracks are propagated over the image, which is not optimal but inevitable if a re-activation of tracks after occlusion should be possible. (b) In the multi-hypothesis approach, the incorrect inactive tracks mistakenly lead to ambiguous assignments. The involved ambiguous detections (marked as purple dotted boxes) increase the set of hypotheses for the incorrect inactive tracks. As a consequence, many incorrect hypotheses emerge that can introduce tracking errors.

One can see that the multi-hypothesis approach for ambiguous assignments with missing detections ($n_D < n_T$) improves IDF1 by 1.7 points, however, MOTA is reduced by 0.6 points. Qualitatively, it is observed that the approach is vulnerable to some cases with incorrect inactive tracks as shown in Figure 4.1. Incorrect inactive tracks always pose a risk for introducing tracking errors. The multi-hypothesis approach increases this risk, when multiple hypotheses for



(a) Baseline

(b) Proposed approach

Figure 4.2: Positive example of the proposed multi-hypothesis approach. (a) In the standard association, the ID of the man with light blue shirt switches with the ID of the occluded man with cap. The cause of the IDSW is the ambiguous detection indicated with a purple dotted box in the lower middle frame. (b) Instead of directly assigning this ambiguous detection, multiple hypotheses (4 in total, for both tracks one with track propagation and one with assigned detection) are built. Later, the assignment task is clear again. Postponing the association decision prevents the IDSW.

such an incorrect inactive track are built. In future experiments, this problem should be further investigated. It might be better to consider only active tracks for building hypotheses. One situation, where the multi-hypothesis approach successfully solves an ambiguous assignment problem can be found in Figure 4.2. Whereas in the baseline association, the ambiguous detection is assigned to the wrong track leading to an identity switch, all targets can be successfully tracked when the assignment decision is postponed with the help of multiple hypotheses

until the situation is clear again. Note that the example in Figure 4.2, where two tracks are competing for one detection, is exactly as depicted in Figure 3.1(a).

Having again a look at Table 4.1, it can be seen that the multi-hypotheses approach for situations with more detections than tracks ($n_D > n_T$) both enhances IDF1 and MOTA by 0.8 and 0.4 points, respectively. As expected, the number of FP can be greatly reduced, since many duplicate detections appear only in single frames and thus are removed with the multi-hypothesis approach as in Figure 3.1(b). Furthermore, the number of IDSW is slightly decreased.

Unfortunately, applying the multi-hypothesis approach for all types of ambiguous assignments (last line in Table 4.1), the results cannot be further improved. Whereas the decrease in MOTA is expected as the MH ($n_D > n_T$) variant also lowers MOTA, the reduction of IDF1 is surprising.

In future works, a deeper analysis has to be made to better understand the negative impact of incorrect inactive tracks in the multi-hypothesis approach. Furthermore, more ablative experiments could be run (on tracking parameters) to get a deeper understanding of the proposed approach and find out other possible error sources. Nevertheless, it has been shown, that separately treating ambiguous assignments is a promising idea, for which the development of other strategies, next to the proposed multi-hypothesis approach, should be explored.

5 Conclusion

In this report, a novel association technique for multi-target tracking is proposed that treats ambiguous assignments separately with a multi-hypothesis approach. The ambiguous assignments are determined purely based on the distance matrix of tracks and detections and thus, the proposed method can be applied in any tracking-by-detection based approach. The track hypotheses allow to postpone the association decision for ambiguous assignments until the situation is clear again, which can improve the association accuracy in ambiguous situations. Besides showing the superiority of the proposed approach in comparison to the standard association in some scenarios, also some weaknesses are identified and suggestions for possible improvements in future works are made.

References

[1] Keni Bernardin and Rainer Stiefelhagen. “Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics”. In: *EURASIP Journal on Image and Video Processing* 2008 (2008).

[2] Alex Bewley et al. “Simple Online and Realtime Tracking”. In: *2016 IEEE International Conference on Image Processing (ICIP)*. 2016, pp. 3464–3468.

[3] Erik Bochinski, Volker Eiselein, and Thomas Sikora. “High-Speed Tracking-by-Detection Without Using Image Information”. In: *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. 2017.

[4] Kaiming He et al. “Deep Residual Learning for Image Recognition”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778.

[5] Harold W. Kuhn. “The Hungarian Method for the Assignment Problem”. In: *Naval Research Logistics Quarterly* 2.1–2 (1955), pp. 83–97.

[6] Tsung-Yi Lin et al. “Feature Pyramid Networks for Object Detection”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 936–944.

[7] Qiankun Liu et al. “GSM: Graph Similarity Model for Multi-Object Tracking”. In: *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*. 2020, pp. 530–536.

[8] Anton Milan et al. “MOT16: A Benchmark for Multi-Object Tracking”. In: *arXiv:1603.00831* (2016).

[9] Shaoqing Ren et al. “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.6 (2017), pp. 1137–1149.

[10] Ergys Ristani et al. “Performance Measures and a Data Set for Multi-target, Multi-camera Tracking”. In: *Computer Vision – ECCV 2016 Workshops*. Vol. 9914. Lecture Notes in Computer Science. 2016, pp. 17–35.

[11] Shuai Shao et al. “CrowdHuman: A Benchmark for Detecting Human in a Crowd”. In: *arXiv:1805.00123* (2018).

[12] Andreas Specker et al. “An Occlusion-Aware Multi-Target Multi-Camera Tracking System”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. 2021, pp. 4173–4182.

[13] Daniel Stadler and Jürgen Beyerer. “Improving Multiple Pedestrian Tracking by Track Management and Occlusion Handling”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 10958–10967.

[14] Daniel Stadler, Lars Wilko Sommer, and Jürgen Beyerer. “PAS Tracker: Position-, Appearance- and Size-Aware Multi-object Tracking in Drone Videos”. In: *Computer Vision – ECCV 2020 Workshops*. Vol. 12538. Lecture Notes in Computer Science. 2020, pp. 604–620.

[15] Siyu Tang et al. “Multiple People Tracking by Lifted Multicut and Person Re-identification”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 3701–3710.

[16] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. “Simple Online and Realtime Tracking with a Deep Association Metric”. In: *2017 IEEE International Conference on Image Processing (ICIP)*. 2017, pp. 3645–3649.

[17] Bin Xiao, Haiping Wu, and Yichen Wei. “Simple Baselines for Human Pose Estimation and Tracking”. In: *Computer Vision – ECCV 2018*. Vol. 11210. Lecture Notes in Computer Science. 2018, pp. 472–487.

[18] Jiarui Xu et al. “Spatial-Temporal Relation Networks for Multi-Object Tracking”. In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 3987–3997.

Conceptualization of a Trust Dashboard for Distributed Usage Control Systems

Paul Georg Wagner

Vision and Fusion Laboratory
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
paul.wagner@kit.edu

Abstract

Achieving data protection and privacy in modern data processing systems is a prominent topic of academic research today. The goal of retaining comprehensive informational sovereignty requires new and innovative solutions, both technological and methodological in nature. Distributed usage control is a popular technology that can give data providers the ability to actively govern the usage of their personal information even in remote systems. However, the architecture of distributed usage control systems is rather complex and often highly dynamic. This makes the assessment of the system's soundness and trustworthiness difficult, especially for untrained laypersons.

In this work we present the concept of a trust dashboard for distributed usage control systems that are backed by trusted computing technologies. The trust dashboard is intended to give users a visual intuition about the current state of the usage control system and its trustworthiness. We achieve this by using a formal model to describe relevant trust dependencies and the actually conducted remote attestations between usage control components, as well as a-priori trust levels for system operators. Based on this we propose a visualization concept that illustrates the current system state and estimates the overall trustworthiness

of the system. Ultimately the trust dashboard aids system operators in the assessment of dynamic and distributed usage control architectures.

1 Introduction

Data privacy is one of the major IT-security challenges of our time. Since we live in a world of ubiquitous data acquisition, keeping track of our personal information is an important even though arduous task. This is especially true when personal data are being distributed in highly interconnected and decentralized systems. In the past years, the notion of *data sovereignty* has become prominent in academic research. While no universally accepted definition of (data) sovereignty exists today, it seems clear that retaining ownership of shared information plays an essential role [5]. Hence on our way to data sovereignty, we need to give individuals the possibility to not just track, but control their own private information wherever it may be stored and evaluated.

One technology that can aid in this task is *usage control*. Usage control allows the specification of access rights and obligations that are continuously enforced on a data set throughout its life cycle [9]. Introduced by Park and Sandhu [10] almost two decades ago, a few usage control variants have been developed since then. For one, *distributed usage control* [11] allows the enforcement of usage rules even across system and domain boundaries. This is achieved by operating several independent usage control components in every participating domain. These components then work together over the network to disseminate all usage rules and enforce them simultaneously on all domain systems. Since distributed usage control components have to safeguard critical data and enforce usage rules even on potentially hostile systems, securing them against manipulation and attacks is by no means a trivial task. Most often the integrity of usage control components is protected using *trusted computing* hardware such as Trusted Platform Modules (TPMs). Trusted computing technologies can provide hardware-backed security guarantees that prevent even system owners such as malicious data receivers from tampering with critical parts of the usage control system. With the development of more powerful usage control models, the resulting policy languages and system implementations became increasingly

complicated. Especially in distributed scenarios with many participants, a multitude of usage control components are required for a reliable enforcement of all usage rules. All of this makes it very hard for data owners or even system administrators to decide if the current configuration of a usage control system is safe and if all usage control components are in an acceptable state.

In this work we explore the possibility of using a trust dashboard to aid system owners in the evaluation of distributed usage control systems. This dashboard is intended to give the users a visual intuition about the current state of the usage control system and its trustworthiness. The remainder of this paper is structured as follows. In section 2 we give a brief introduction of distributed usage control systems and describe how they can be protected with trusted computing technologies. Afterwards in section 3 we present a suitable model capable of expressing trust in distributed usage control systems. Based on this model, in section 4 we then define goals and requirements for a trust dashboard and propose a concept for visualizing various states of the usage control system in an intuitive way. We conclude this paper in section 5 with an outlook on future work on this research question.

2 Related Work

To prepare for the description of our trust dashboard, we first give a brief introduction of distributed usage control systems and show how trusted computing technologies are typically used to protect them against malicious tampering.

2.1 Distributed Usage Control

Usage control (UC) was first proposed in 2002 as a generalization of traditional access control methods [10]. A formal usage control model has been published in 2004 by Park and Sandhu [9]. In general, usage control allows for continuous authorization of data accesses even if the data are already in use. This is in contrast to classical access control schemes, where authorization decisions are made only at the time of the initial data access. Furthermore, usage control supports the declaration of obligations that need to be fulfilled before, during or

after a certain data usage. This is not covered by classical access control either. Ultimately usage control methods can help to describe and enforce complex data usage strategies, such as limiting the number of views or the time of access to sensitive information.

Based on the original usage control model by Park and Sandhu [9], several model variants and formalizations have been developed since [9, 11, 18, 7, 8, 6] One of the most influential improvements is distributed usage control (DUC). Developed by Pretschner et al. [11] in 2006, distributed usage control deals with the enforcement of usage rules even across system and domain boundaries. It allows data providers to specify usage policies in a domain-specific language [4], which are then distributed alongside the sensitive information to any remote data receivers. One way of implementing distributed usage control systems is to rely on a derivative of the XACML reference architecture [13]. Originally developed for attribute-based access control, the XACML components can be canonically extended to implement usage control policies. Figure 2.1 shows a distributed usage control system based on XACML components.

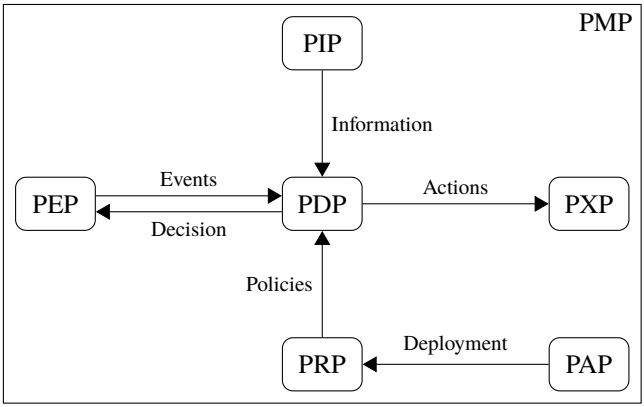


Figure 2.1: Distributed usage control components.

At the heart of any distributed usage control system is the *policy enforcement point (PEP)*. PEPs are usually implemented close to data processing applications and are capable of continuously examining and influencing data accesses. Based

on this examination, PEPs broadcast notifications of data usage requests into the usage control system. These notifications are then received and processed by a suitable *policy decision point (PDP)*. PDPs create usage control decisions by evaluating the received notifications against the set of active usage control policies. In addition to the classical binary access decision of `allow` versus `deny`, the PDP can also rule that the data usage described by the event should be `modified` prior to its execution. In the end the PEP receives the decision from its PDP and enforces it on the data processing application.

To aid the PDP in creating correct usage control decisions, two more usage control components are required. The *policy information point (PIP)* can be queried by the PDP for subject and object attributes, as well as generic information such as database entries or environmental properties. The *policy execution point (PXP)* is responsible for executing obligations demanded prior to a data usage, for example the incrementation of an access counter. Obligations are invoked by the PDP and have to be executed successfully before the PDP publishes a positive decision. By utilizing PIP and PXP capabilities, complex and expressive usage control policies can be specified and enforced.

Furthermore there are three auxiliary components involved in the usage control enforcement process. The *policy administration point (PAP)* provides data owners and system administrators an interface to specify and manage usage control policies for their respective data sets. The *policy retrieval point (PRP)* is used by PDPs to retrieve policies from remote usage control systems, e.g. if data accesses to external resources are requested. Finally, the *policy management point (PMP)* controls the distributed usage control system, aids in the lookup of usage control components and facilitates establishing connections between them. In the end it is the collaboration of all components that ensures proper usage control enforcement. Even though the original XACML architecture was intended to specify logical instead of physical system components, in the case of distributed usage control these components are running as dedicated services on different computer systems.

2.2 Trusted Computing

Distributed usage control systems allow data providers to restrict access to critical information even after it has been released. However, this only works under the assumption that the participating usage control components are all working correctly. Especially if multiple usage control systems operated by different stakeholders are involved, this assumption is not necessarily sound. For example, remote data receivers may be motivated to tamper with their own usage control components to bypass the enforcement of transmitted policies. Because of this, technical measures have to be taken to prevent malicious system operators from tampering with distributed usage control components.

The proposed solution to this problem is making use of *trusted computing* technologies [3, 2]. Trusted computing allows to protect running software from external influences and verify their integrity by means of hardware-based access control. Among the most widespread trusted computing technologies today are Trusted Platform Modules (TPMs). Usually TPMs consist of a dedicated hardware chip that has been manufactured according to a specification developed by the Trusted Computing Group (TCG) [14]. Many modern desktop and server motherboards already include a TPM hardware chip soldered onto the PCB. Even if no physical TPM is available, software TPMs can be included in the device firmware [12]. However, these firmware modules obviously do not provide the same level of security as their hardware-based counterparts. In general, TPMs are designed to create and hold several cryptographic keys in hardware, which can then be used to encrypt and sign critical information in a secure environment. The private parts of the cryptographic keys stored in the TPM hardware are protected against external influence and cannot be extracted in plain text. Furthermore, TPMs offer *remote attestation* functionality. Remote attestation allows external verifiers to uniquely identify a TPM-equipped computer system and attest to the current state of its software stack. To achieve this, the TPM is used to store unforgeable fingerprints of the current hardware and software configuration in a special set of registers. Then a remote verifier can probe the measured system for proof of its current system state. This is usually done via a cryptographic protocol [17], which establishes a secure channel to the system under test and transmits a so-called *quote*. The quote is a

data structure containing the current system fingerprints and is cryptographically signed by the TPM. The verifier then validates the correctness of the signature and cross-checks the attested fingerprints inside the quote with a set of expected values. If everything validates correctly, the verifier is convinced that the attested fingerprints are correct and the remote system indeed runs a correctly configured and unmodified software stack.

TPM-based remote attestation allows data providers to verify the integrity of remote usage control components before trusting them with enforcing usage rules on their critical data. However, some drawbacks of using TPMs for this purpose have been discovered as well, which can be mitigated by using more powerful trusted computing technologies such as Intel SGX [16]. In any case, conducting remote attestations remains the fundamental instrument of establishing trust in distributed usage control systems.

3 Modeling Trust in Usage Control Systems

In order to estimate the trustworthiness of distributed usage control systems, we first require a suitable model. This model should describe existing trust dependencies between usage control components as well as the individually conducted remote attestations at any point in time. For this we partially rely on a model that has been published as part of our previous work [15]. However, for the purpose of defining a trust dashboard we have to extend this base model with the possibility to distinguish different levels of trust and express the trustworthiness of system operators. In the remainder of this section we briefly present the relevant parts from the original publication [15] and then extend the base model to meet our requirements.

3.1 Base Model

As described in section 2.1, the basis of a distributed usage control architecture is formed by a set of usage control modules M (e.g. PEP, PDP, ...). Furthermore, we define a set of usage control functions F (e.g. deploy, evaluate, ...). The basic semantic of distributed usage control systems is then specified via a *trust*

dependency graph $T = (M, E_T, l_T)$. The edges $E_T \subseteq M \times M$ of this graph describe the trust relationships between the different types of usage control components. More concretely, an edge $(u, v) \in E_T$ means that usage control component u requires an honest component v to perform its task correctly. Finally, the mapping $l_T : E_T \to F$ assigns a human-readable label to each trust dependency, depending on which usage control function is responsible for the dependency. Figure 3.1 shows the trust dependency graph for the XACML-based distributed usage control system presented in section 2.1.

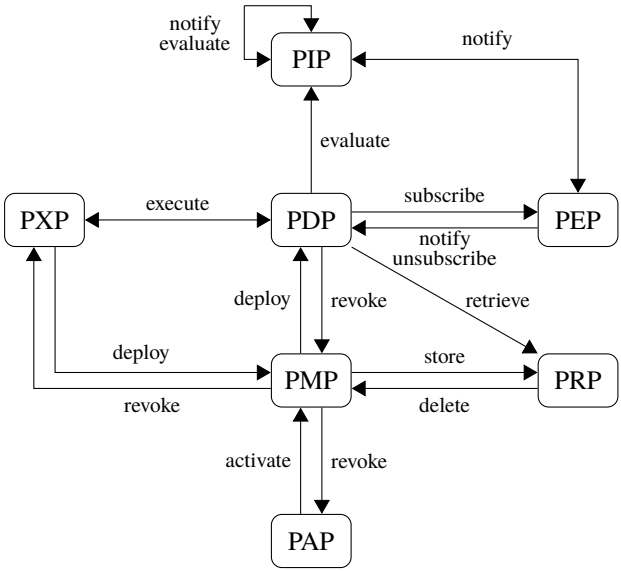


Figure 3.1: Trust dependency graph [15].

From the trust dependency graph, an *instance graph* $I = (V_I, E_I, l_I, type_I)$ can be derived. This graph contains concrete instances V_I of usage control components that are being executed, as well as the appropriate trust dependencies E_I and label mappings l_I from the trust dependency graph. Finally, the mapping $type_I : V_I \to M$ assigns a module type to each concrete component instance.

While the trust dependency graph describes the logical architecture of the usage control system, the instance graph describes a concrete realization of it.

Furthermore, we partition the vertices of the instance graph into disjunctive sets called *attestation containers*, denoted by $C \subseteq V_I$. An attestation container describes a set of usage control modules that can be jointly attested. Which module instances form an attestation container depends on the used attestation technology and the system architecture (c.f. section 2.2). The set of all attestation containers is denoted by $\mathcal{C} \subseteq \mathcal{P}(V_I) \setminus \emptyset$. For convenience purposes, we define a function $c_I : V_I \to \mathcal{C}$ that maps a given component to its attestation container. Figure 3.2 shows an example of an instance graph with attestation containers.

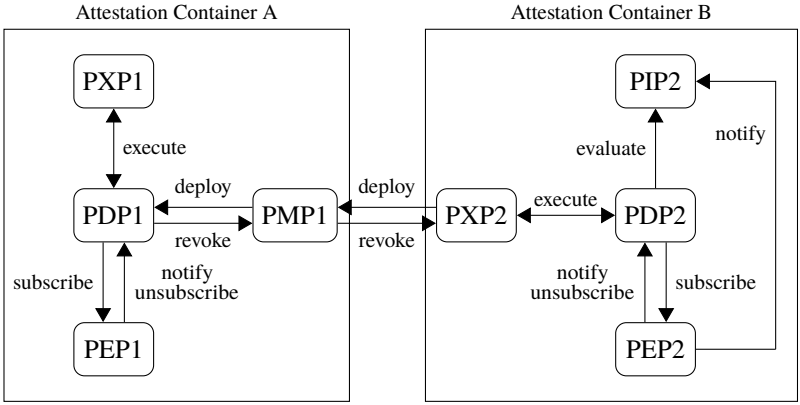


Figure 3.2: Instance graph with attestation containers.

Finally, for each component instance $v \in V_I$ we can define the *attestation schedule* $att_v : \mathbb{N}^+ \times \mathcal{C} \to \{-1, 0, 1\}$. The attestation schedule describes the remote attestations that the component v conducts at a certain point in time, and if they are successful (1), not attempted (0) or not successful (-1). For more details as well as some examples of the base model we refer the reader to the original publication [15].

3.2 Extended Model

To use this model as basis for a trust dashboard, we first extend it with the concept of *operators*. Operators are running usage control components in their own infrastructure and are participating in the distributed usage control system. They can be seen as the administrators responsible for managing the usage control components of their company, or even conceptually as the entire organization itself. Operators can act both as data provider and data receiver, respectively releasing or collecting sensitive information as well as associated usage control policies. We denote the set of operators with $\mathcal{O}$. For each attestation container $C \in \mathcal{C}$ we then identify a single operator $O \in \mathcal{O}$ that is responsible for all usage control components running in the attestation container C. We describe this with the operator mapping $o : \mathcal{C} \to \mathcal{O}$. While each attestation container is managed by exactly one operator, a single operator can be responsible for multiple attestation containers at once.

Furthermore, we want to associate a certain level of trust with each operator, and subsequently with the usage control components they are running. For this we define four different trust levels `full`, `marginal`, `untrustworthy` and `unknown`. This categorization is inspired by the trust level definition of PGP [1]. We denote the set of trust levels with $\mathcal{T}$. Finally, we can assign each operator a trust level with the mapping $t : \mathcal{O} \to \mathcal{T}$. This mapping will be customizable by the user of the trust dashboard and serve as basis for including subjective trust assessments into the estimations provided by the dashboard.

4 Conceptualizing a Trust Dashboard

In this section we propose a concept for a trust dashboard that helps users to gain insight into the current state of a distributed usage control system by means of an intuitive visual representation. This dashboard is mainly intended to be used by direct participants of the distributed usage control system, such as system operators and/or data providers. The trust dashboard is supposed to assist them in deciding if the distributed system components can be deemed trustworthy, before issuing critical usage control policies or releasing sensitive data.

4.1 Goals and Requirements

To present our proposed trust dashboard, we first define the two main design goals of the dashboard and derive appropriate requirements for each of them.

Goal 1: Visualizing the system state. The first main goal of our trust dashboard is to give the user an intuitive overview of the current usage control system state. For this, we need to show the user what usage control components are currently running and what dependencies exist between them. Furthermore, we need to illustrate what remote attestations have already been conducted and point out what parts of the trust dependencies have been covered as a result. Ultimately we identify three requirements to achieve this goal.

(Req. 1) Show the user the set of active distributed usage control components, both local and remote (i.e. operated by other participants).

(Req. 2) Show the user the required trust dependencies between the active usage control components.

(Req. 3) Highlight to the user any discrepancies between the required trust dependencies and the actually conducted remote attestations.

Goal 2: Visualizing the system trustworthiness. The first goal only aims at expressing the current state of the distributed usage control system in an intuitive way. However, for the trust dashboard this objective alone is not sufficient. We also have to give the user feedback about the resulting trustworthiness of the usage control system in its current state. More concretely, we need to allow the user to specify their subjective estimation of the trustworthiness of certain system operators. Obviously this is highly dependent on the application context, e.g. if a remote system operator is a competitor of the trust dashboard user. Based on this we can then offer some estimates about the trustworthiness of the system overall. In the end goal 2 yields two more requirements to be considered.

(Req. 4) Allow the user to specify their a-priori level of trust in the operators of remote usage control systems.

(Req. 5) Show the user a qualitative estimation of the overall trustworthiness of the usage control system, based on the current attestation schedule and the a-priori trust levels specified by the user.

The remainder of this section shows how we achieve the identified goals and requirements in our proposed dashboard. Finally we illustrate the resulting visual representations of the different trust dashboard states in a small example.

4.2 Goal 1: Visualizing the System State

In order to conceptualize a trust dashboard for distributed usage control, we translate the model from section 3 into a simple visualization of the current system state. Starting with an instance graph $I = (V_I, E_I, l_I, type_I)$, as a first step the currently active usage control components V_I are displayed as boxes, labeled with the respective module names derived from the mapping $type_I$. Similarly, the set of operators $\mathcal{O}$ is represented by a number of larger, unfilled boxes around the usage control components. Which components are displayed in which operator boxes is determined by the operator mapping $o : \mathcal{C} \to \mathcal{O}$, depending on the attestation container that the component in question resides in. Together, this satisfies requirement 1. Furthermore, the trust dependency edges E_I between usage control components are visualized with directed arrows connecting the component boxes in the direction of the trust dependency (requirement 2). The states of the trust dependency edges are derived from the current attestation schedules $att_v : \mathbb{N}^+ \times \mathcal{C} \to \{-1, 0, 1\}$ for each usage control component $v \in V_I$ and visualized using different colors for the arrows (requirement 3). Depending on the current time step t and the attestation schedules, we distinguish four different states that a trust dependency edge can be in.

Recently verified dependency. A trust dependency edge $(u, v) \in E_I$ is recently verified at time t, if there is a successful attestation between u and v not older than $\bar{t}$, and no failed attestation has occurred since. The time interval $\bar{t}$ is a previously defined constant that represents how long a component should be fully trusted after a successful attestation. We mark recently verified trust

dependencies with a green arrow in the trust dashboard. In the formal model, this concept is expressed as equation 4.1.

$$\text{color}(t,u,v) = \text{green} \iff \begin{array}{l} \exists t_s > (t-\bar{t}) : att_u\left(t_s, c_I\left(v\right)\right) = 1 \\ \wedge\, \nexists t_f > t_s : att_u\left(t_f, c_I\left(v\right)\right) = -1 \end{array} \tag{4.1}$$

Formerly verified dependency. A trust dependency edge $(u,v) \in E_I$ is formerly verified at time t, if there is a successful attestation between u and v older than $\bar{t}$, and neither a failed nor a successful attestation has occurred since. We mark formerly verified trust dependencies with a yellow arrow in the trust dashboard. In the formal model, this concept is expressed as equation 4.2.

$$\text{color}(t,u,v) = \text{yellow} \iff \begin{array}{l} \exists t_s \leq (t-\bar{t}) : att_u\left(t_s, c_I\left(v\right)\right) = 1 \\ \wedge\, \forall \tilde{t} > t_s : att_u\left(\tilde{t}, c_I\left(v\right)\right) = 0 \end{array} \tag{4.2}$$

Unverified dependency. A trust dependency edge $(u,v) \in E_I$ is unverified at time t, if there are no attestations between u and v so far. We mark unverified trust dependencies with a black arrow in the trust dashboard. In the formal model, this concept is expressed as equation 4.3.

$$\text{color}(t,u,v) = \text{black} \iff \forall \tilde{t} \leq t : att_u\left(\tilde{t}, c_I\left(v\right)\right) = 0 \tag{4.3}$$

Invalidated dependency. A trust dependency edge $(u,v) \in E_I$ is invalidated at time t, if there is a failed attestation between u and v and no successful attestation has occurred since. We mark invalidated trust dependencies with a red arrow in the trust dashboard. In the formal model, this concept is expressed as equation 4.4.

$$\text{color}(t,u,v) = \text{red} \iff \begin{array}{l} \exists t_f \leq t : att_u\left(t_f, c_I\left(v\right)\right) = -1 \\ \wedge\, \nexists t_s > t_f : att_u\left(t_s, c_I\left(v\right)\right) = 1 \end{array} \tag{4.4}$$

4.3 Goal 2: Visualizing the System Trustworthiness

Based on the visualization of the current system state, an estimation for its overall trustworthiness should be given. This is done in two steps. First, the trust

dashboard users define their subjective level of trust in the various operators of usage control components (requirement 4). Similar to before, this is visualized by coloring the operator boxes depending on the assigned level of trust. As described in section 3.2, we distinguish four different trust levels $\mathcal{T}$. We associate the trust level `full` with the color green, trust level `marginal` with the color yellow, trust level `untrustworthy` with the color red and trust level `unknown` with the color black. In terms of the formal model, this step is defining the operator trust mapping $t : \mathcal{O} \to \mathcal{T}$.

Afterwards the trust levels in individual usage control components have to be derived. This is done based on the current state of the trust dependency edges as described in the previous section. For the sake of consistency we use the same trust level categorization for the usage control components as for the operators. Furthermore, the component boxes in the trust dashboard are colored in the same manner as the trust dependency edges and the operator boxes.

Fully trusted component. A usage control component $v \in V_I$ is fully trusted at time t if there is at least one recently verified and no invalid trust dependency to v. We mark fully trusted components with a green box in the trust dashboard. In the formal model, this concept is expressed as equation 4.5.

$$\text{color}(t, v) = \text{green} \iff \begin{array}{l} \exists u \in V_I : \text{color}(t, u, v) = \text{green} \\ \wedge \nexists u \in V_I : \text{color}(t, u, v) = \text{red} \end{array} \tag{4.5}$$

Marginally trusted component. A usage control component $v \in V_I$ is marginally trusted at time t if there is at least one formerly verified trust dependency to v, but no recently verified or invalid ones. We mark marginally trusted components with a yellow box in the trust dashboard. In the formal model, this concept is expressed as equation 4.6.

$$\text{color}(t, v) = \text{yellow} \iff \begin{array}{l} \exists u \in V_I : \text{color}(t, u, v) = \text{yellow} \\ \wedge \forall u \in V_I : (\text{color}(t, u, v) = \text{yellow} \\ \vee \text{color}(t, u, v) = \text{black}) \end{array} \tag{4.6}$$

Untrusted component. A usage control component $v \in V_I$ is untrusted at time t if there is at least one invalid trust dependency to v. We mark untrusted components with a red box in the trust dashboard. In the formal model, this concept is expressed as equation 4.7.

$$\text{color}(t, v) = \text{red} \iff \exists u \in V_I : \text{color}(t, u, v) = \text{red} \tag{4.7}$$

Component with unknown trust level. A usage control component $v \in V_I$ has an unknown trust level at time t if there are only unverified trust dependencies to v. We mark components of unknown trust level with a black box in the trust dashboard. In the formal model, this concept is expressed as equation 4.8.

$$\text{color}(t, v) = \text{black} \iff \forall u \in V_I : \text{color}(t, u, v) = \text{black} \tag{4.8}$$

Once the operator trust mapping and the trust level of individual components is defined, an overall trust estimation should be derived from it (requirement 5). For simplicity we use a qualitative estimation with three trust levels that are each visualized with a distinct icon in the trust dashboard.

Trusted overall system state. A distributed usage control system with instance graph I is in a trusted state at time t if all usage control components of I are either fully trusted, or belong to a system operator that is fully trusted by the trust dashboard user. Broadly speaking, this means that the usage control system has verified all doubtful components with a recently conducted remote attestation. In the trust dashboard, this overall system state is visualized by a green checkmark next to the trust graph. In the formal model, it is expressed as equation 4.9.

$$\text{state}(t) = \text{trusted} \iff \begin{array}{c} \forall v \in V_I : \text{color}(t, v) = \text{green} \\ \vee\, t\left(o\left(c_I\left(v\right)\right)\right) = \text{full} \end{array} \tag{4.9}$$

Ambiguous overall system state. A distributed usage control system with instance graph I is in an ambiguous state at time t if it is not trusted, but no untrusted components exist in I either. This means that not every potentially dangerous usage control component has been cryptographically verified, but

there is no evidence for malicious behavior. In the trust dashboard, this overall system state is visualized by a yellow questionmark next to the trust graph. In the formal model, it is expressed as equation 4.10.

$$\text{state}(t) = \text{ambiguous} \iff \begin{array}{l} \text{state}(t) \neq \text{trusted} \\ \wedge \forall v \in V_I : \text{color}(t, v) \neq \text{red} \end{array} \tag{4.10}$$

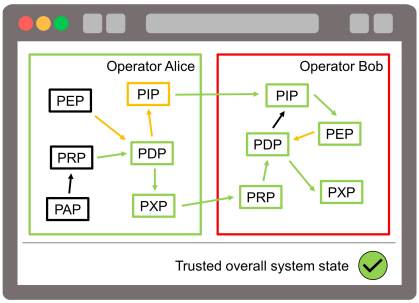
Untrusted overall system state. A distributed usage control system with instance graph I is in an untrusted state at time t if there is at least one untrusted component in I. This means that at least one usage control component has failed the verification via a remote attestation, and hence we have to assume malicious interference in the usage control system. In the trust dashboard, this overall system state is visualized by a red crossmark next to the trust graph. In the formal model, it is expressed as equation 4.11.

$$\text{state}(t) = \text{untrusted} \iff \exists v \in V_I : \text{color}(t, v) = \text{red} \tag{4.11}$$

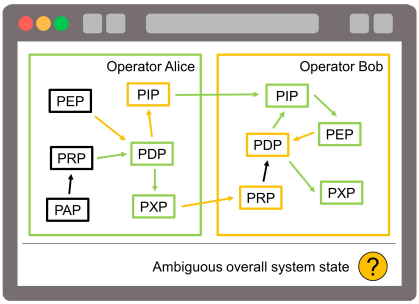
4.4 Trust Dashboard Examples

To illustrate the nature of the resulting visualization, in figure 4.1 we give a few simple examples of the trust dashboard for all three system states. For this we assume there to be two operators Alice and Bob, each managing several usage control components. In our examples we show the trust dashboard from the point of view of operator Alice, so we assume her to always be a fully trusted operator. As described in section 4.3, this is indicated by a green operator box around her components.

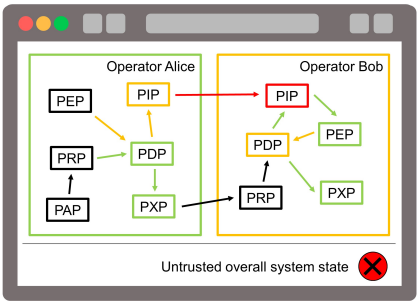
The first example in figure 4.1(a) shows the dashboard layout in a trusted overall system state. This is despite operator Bob being untrusted in this example, as indicated by the red operator box around his components. Since all of Bob's components have at least one recently verified incoming dependency (green arrows), they are all classified as fully trusted and marked with green boxes (see eq. 4.5). As a result, all components of the system are either fully trusted or belong to a fully trusted operator. With eq. 4.9 follows that the overall system state is trusted. In the second example in figure 4.1(b), Bob's PRP only has a



(a) Trusted system state.



(b) Ambiguous system state.



(c) Untrusted system state.

Figure 4.1: Examples of the trust dashboard visualization in the three system states.

formally verified dependency (yellow arrow), most likely because the previously conducted attestation timed out. Because of this, Bob's PRP is left as only marginally trusted (see eq. 4.6) and the overall system state results to ambiguous, as described in eq. 4.10. The last example in figure 4.1(c) shows an invalidated dependency between Alice's PIP and Bob's PIP (red arrow). This invalidation is due to a failed remote attestation between both PIPs, which results in Bob's PIP being classified as untrusted (see eq. 4.7) and marked with a red box. Because of this deterioration in trust, according to eq. 4.11 now the entire system has to be seen as untrustworthy as well.

5 Conclusion

In this work we presented the concept of a trust dashboard for distributed usage control systems that are backed by trusted computing technologies. Based on a formal model describing generic usage control systems and the relevant trust dependencies, we proposed a visualization concept that (i) illustrates the current system state and (ii) estimates the overall trustworthiness of the system. To achieve the first goal, we classified the trust dependencies between distributed usage control components based on the recency and outcome of conducted remote attestations. By combining the current system state with a-priori trust levels for system operators, we then provided the dashboard user with a qualitative estimation of the overall system trustworthiness. Ultimately our approach contributes a tool to help usage control system operators in the assessment of dynamic and distributed system architectures.

In the future we plan to implement our concept as a web-based application and deploy it to real-world usage control systems. Based on this, user studies can be conducted to evaluate the benefits of the trust dashboard in real-world scenarios. Furthermore we are aware of some issues that remain unaddressed with our approach. So far the specific trusted computing technologies protecting the usage control components are not being considered in either the formal model or the trust dashboard concept. However, since different technologies yield different benefits and drawbacks, clearly there are opportunities to refine and improve both the visualization of the current system state, as well as the

resulting trust estimation. In addition, the current trust estimation method is based on a simple qualitative heuristic. To obtain better trust estimation results, a more sophisticated approach based on probabilistic estimations can be undertaken. This would result in a quantitative (albeit still subjective) trust estimation methodology.

References

[1] Alfarez Abdul-Rahman. "The pgp trust model". In: *EDI-Forum: the Journal of Electronic Commerce*. Vol. 10. 3. 1997, pp. 27–31.

[2] Masoom Alam et al. "Model-based behavioral attestation". In: *Proceedings of the 13th ACM symposium on Access control models and technologies*. 2008, pp. 175–184.

[3] Agreiter Berthold et al. "A technical architecture for enforcing usage control requirements in service-oriented architectures". In: *Proceedings of the 2007 ACM workshop on Secure web services*. 2007, pp. 18–25.

[4] Manuel Hilty et al. "A policy language for distributed usage control". In: *European Symposium on Research in Computer Security*. Springer. 2007, pp. 531–546.

[5] Patrik Hummel et al. "Data sovereignty: A review". In: *Big Data & Society* 8.1 (2021), p. 2053951720982012.

[6] Radha Jagadeesan et al. "Timed constraint programming: a declarative approach to usage control". In: *Proceedings of the 7th ACM SIGPLAN international conference on Principles and practice of declarative programming*. 2005, pp. 164–175.

[7] Helge Janicke, Antonio Cau, and Hussein Zedan. "A note on the formalisation of UCON". In: *Proceedings of the 12th ACM symposium on Access control models and technologies*. 2007, pp. 163–168.

[8] Fabio Martinelli and Paolo Mori. "A Model for Usage Control in GRID systems". In: *2007 Third International Conference on Security and Privacy in Communications Networks and the Workshops-SecureComm 2007*. IEEE. 2007, pp. 169–175.

[9] Jaehong Park and Ravi Sandhu. “The UCON ABC usage control model”. In: *ACM Transactions on Information and System Security (TISSEC)* 7.1 (2004), pp. 128–174.

[10] Jaehong Park and Ravi Sandhu. “Towards usage control models: beyond traditional access control”. In: *Proceedings of the seventh ACM symposium on Access control models and technologies*. 2002, pp. 57–64.

[11] Alexander Pretschner, Manuel Hilty, and David Basin. “Distributed usage control”. In: *Communications of the ACM* 49.9 (2006), pp. 39–44.

[12] Himanshu Raj et al. “ftpm: A software-only implementation of a {TPM} chip”. In: *25th {USENIX} Security Symposium ({USENIX} Security 16)*. 2016, pp. 841–856.

[13] OASIS Standard. *extensible access control markup language (xacml) version 3.0*. 2013.

[14] *TCG Specification Architecture Overview. Specification Revision 1.4*. `https://trustedcomputinggroup.org/wp-content/uploads/TCG_1_4_Architecture_Overview.pdf`. Accessed: 2021-11-16.

[15] Paul Georg Wagner. “Towards a Formal Model for Quantifying Trust in Distributed Usage Control Systems”. In: *Proceedings of the 2019 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory*. 2019, p. 113.

[16] Paul Georg Wagner, Pascal Birnstill, and Jürgen Beyerer. “Distributed usage control enforcement through trusted platform modules and sgx enclaves”. In: *Proceedings of the 23nd ACM on Symposium on Access Control Models and Technologies*. 2018, pp. 85–91.

[17] Paul Georg Wagner, Pascal Birnstill, and Jürgen Beyerer. “Establishing Secure Communication Channels Using Remote Attestation with TPM 2.0”. In: *International Workshop on Security and Trust Management*. Springer. 2020, pp. 73–89.

[18] Xinwen Zhang. *Formal model and analysis of usage control*. George Mason University, 2006.

Cross-Domain Fine-Grained Classification: A Review

Stefan Wolf

Vision and Fusion Laboratory
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
stefan.wolf@kit.edu

Abstract

Fine-grained classification is an interesting but challenging task due to the high amount of data needed to achieve a high accuracy. However, the high specificity of the classes makes it difficult to collect a large amount of samples. Thus, the use of cross-domain learning is an interesting aspect since an abundant amount of data exists for some domains like web images exists. In this review, the current works of cross-domain fine-grained classification are summarized and potential areas for future work are highlighted. Even though first works exist, the variety of methods is still small and interesting cross-domain settings are rarely considered. Thus, the field of cross-domain fine-grained classification provides a large room for future research.

1 Introduction

Fine-grained image classification is a task which has gained attention in recent years due to the application of convolutional neural networks achieving promising results. Another reason for the gained attention is that the applications of fine-grained classification are manifold. Gebru et al. [12] predict the model of

vehicles in order to visually estimate the income of regions in the US. Similar approaches could be applied on parking areas of supermarkets to estimate the income of customers. Usuyama et al. [30] apply fine-grained image classification to visually identify pills in order to reduce the risk of medication errors.

Compared to regular image classification, fine-grained classification induces a lower inter-class variance with often only small details distinguishing classes while view and appearance of images can be highly different within classes resulting in a high intra-class variance. Thus, it is a more difficult task than coarse-grained image classification. Additionally, the high specificity of classes limits the availability of images and increases the required knowledge of annotators. Another aspect making the application of fine-grained classification difficult is that the classes of datasets are usually very specific to a certain region and a certain timeframe. For example, the distribution of cars differs heavily for different regions like Europe and Asia and new car models are constantly introduced rendering available datasets quickly outdated.

However, this can be compensated by crawling images from the web or creating synthetic images by rendering 3D models of cars or pills. While these approaches can create a large amount of labeled data, they often induce a large domain gap since images from the web are usually more polished than images in real-world applications like surveillance and synthetic images do not reach the realism of camera images. However, domain adaptation can support the learning process to enable the use of data from different domains than the targeted domains. Since these image sources are also useful for normal image classification, a broad range of literature is dedicated towards domain adaption for coarse-grained classification as summarized by Wang and Deng [32].

However, cross-domain fine-grained classification introduces new challenges like the small inter-class variance compared to the large inter-domain variance which requires careful consideration during adaptation [41]. Moreover, the high specificity of fine-grained classes brings up cross-domain scenarios which are uncommon for regular image classification like a supervised partially zero-shot scenario. In this case, some classes in the target domain do not have images at all while all other classes have abundant images with annotations available in the target domain [30]. In contrast, classic domain adaptation usually assumes

the availability of all classes in the target domain, even though all data in the target domain is unlabeled [9].

Since existing surveys about fine-grained classification [35, 14] or cross-domain classification [32] do not consider the works combining both fields, this survey gives an overview over the existing works about cross-domain fine-grained classification.

The remainder of this work is structured as follows: section 2 summarizes the literature targeting fine-grained classification while section 3 shortly describes existing works regarding cross-domain classification. Section 4 gives a more detailed view on the intersection of both research areas combining cross-domain and fine-grained classification. Section 5 summarizes the content of this work and highlights research aspects which are interesting for future work.

2 Fine-Grained Classification

Since fine-grained classification shares many of its challenges with usual image classification as mostly investigated on the ImageNet [6] dataset, current deep learning models proven on ImageNet have been established as a good starting point for fine-grained classification [31]. However, the high intra-class variance compared to the low inter-class variance of fine-grained classification tasks motivate the exploration of adaptations. In this regard, multiple authors have found ways to improve the performance of deep-learning models when they are exposed to fine-grained classification tasks. The development branches can be roughly categorized as part-based models, bilinear CNNs, multi-task learning, hierarchical classification, metric learning, temporal classification and webly-supervised approaches.

Part-based models. While localizing discriminating parts has not been widely adopted since CNNs are used for classification, fine-grained classification is an area where part-based classification can still be advantageous. A reason is that the distinguishing regions might not be automatically determined during training because of the datasets being too small for the problem at hand. Thus, multiple authors have approached the integration of part-based classification

schemes with CNNs to improve the accuracy [37, 8]. Huang et al. [15] use a part-based model to provide an interpretation of the classification to the user. However, hand-engineered part models suffer from being not optimal for classification and requiring a high annotation effort. Thus, Simon and Rodner [25] propose an unsupervised part-based model using feature map activations to find discriminating parts.

Bilinear CNNs. Following the idea of part-based models that localization of discriminating parts and creating features should be separated, Lin, RoyChowdhury, and Maji [22] propose a two-stream architecture consisting of two CNNs that combines the final feature maps of both networks with a bilinear module. The bilinear module calculates the outer product of both feature vectors for each pixel of the feature map. The expectation of the authors is that one network locates discriminating parts while the other network extracts discriminating features. Due to the high computational demands of the high dimensional outer product of both feature maps, adaptations have been proposed to reduce its dimension [10, 19]. Yu et al. [40] extend the bilinear CNN scheme by applying cross-layer bilinear pooling, i.e., they pool bilinear features between layers of the network instead of only pooling bilinear features after the last layer.

Multi-task learning. In multi-task learning, an auxiliary task is additionally solved to the main task with the auxiliary task being related to the main task. Due to the relation, it is expected that the auxiliary task supports solving the main task. Depending on the method, the solving of the auxiliary task might be either limited to the training process [3] or might also be performed during inference [26, 23]. To enhance the quality of fine-grained class predictions, Sochor, Herout, and Havel [26] feed automatically extracted 3D bounding boxes of the cars as additional information to the network in order to support the network regularizing the perspective. Chen, Liu, and Yu [3] propose a similar approach by predicting the viewpoint of the image as an auxiliary task. Providing knowledge about the viewpoint during training supports the network in coping with the large intra-class variance due to the high variation regarding viewpoints. Lin et al. [23] propose an approach that fits a 3D model on the image, exploits the localization of object parts to extract features at constant locations, and uses

theses features to perform a classification. Due to repetitively applying this scheme, the correct 3D model for the vehicle model is chosen from a database resulting in a higher classification accuracy.

Hierarchical classification. Fine-grained classes are usually part of a more complex class hierarchy. For example, while fine-grained bird classification is commonly done on the level of species, each species is part of a certain genus which is part of a certain family of birds. Cars are often classified on the level of the year a certain iteration of a model was presented. However, this is part of a hierarchy containing the model and the manufacturer. In the case of cars, a second hierarchy can be built by assigning each model to a certain type of car like van or sedan. Hierarchical classification can be seen as a special form of multi-task learning since the classification of coarse-grained categories is used as an auxiliary task to improve the accuracy of the fine-grained classification. Huo et al. [16] exploit these hierarchies by training multiple layers of the hierarchy in a round-robin manner. Buzzelli and Segantin [2] train cascaded classifiers to reduce the number of classes per classifier.

Metric learning. CNNs for classification usually apply a linear layer combined with a softmax activation on the last feature layer to generate the output probability distribution. During training, the backpropagation algorithm is applied which mostly finds an adequate embedding for the final feature layer that tends to minimize intra-class variance and maximize inter-class variance. However, for fine-grained classification tasks, an explicit loss formulation that increases distance between classes and decreases the distance between features of the same class is commonly applied as alternative (using a kNN-classifier for inference) or additional to a softmax-based classifier [27, 18, 39]. Particularly for a high amount of classes as common in fine-grained classification, metric learning proved advantageous [39].

Temporal classification. Object classification is mostly performed on still images. While a single image is usually sufficient for coarse-grained classification, the discriminating parts for fine-grained classification are often only visible in specific perspectives. Thus, Zhu et al. [44] and Alsahafi et al. [1]

investigated fine-grained classification on videos. Zhu et al. [44] regard the redundant information in videos as the main challenge and propose an approach that combines the feature maps from multiple images and processes the feature maps in an iterative manner. In each step, redundant information from previous steps is suppressed while discriminating parts are attended to. Alsahafi et al. [1] use an object detector to extract accurate crops of the vehicle for each image and combine the per-image classification results by averaging.

Webly-supervised. The amount of samples per class is scarce for fine-grained datasets compared to coarse-grained datasets like ImageNet [6] due to the specificity of classes and the difficulty of labeling. Therefore, multiple authors use community provided image collections in the web which have labels available like, e.g., Flickr. With the class name as query large amounts of data can be gathered [36, 7].

Other works. Some works have investigated methods which have not yet brought up a new branch of research or they consider aspects uncommonly explored. Touvron et al. [28] propose a method called Grafit that enables fine-grained classification based on a training dataset only containing coarse-grained labels. This is achieved by combining an instance loss and a kNN loss in order to learn a fine-grained feature embedding. Cui et al. [5] propose a new training scheme for long-tailed class distributions with a small number of frequent classes and a large number of rare classes. Moreover, the authors propose a domain similarity metric to find a good dataset for pre-training a network prior to training it on the target dataset. Zhang et al. [42] follow an ensemble strategy by training multiple expert networks with the maximization of the Kullback-Leibler divergence between the output probability distributions of each classifier as additional optimization target.

3 Cross-Domain Classification

In a cross-domain classification setting, at least two domains are involved called source and target with the evaluation being performed on data of the

target domain. While abundant data is available for the source domain, the availability of data in the target domain is limited in some form. Mostly the limitation is in form of missing labels. In this case, the adaptation for the target domain is called unsupervised domain adaptation. More settings in the context of fine-grained classification are described in section 4. Formally, cross-domain classification can be described with two sets of images X and labels Y called (X_s, Y_s) and (X_t, Y_t) for source and target, respectively. In a domain adaptation setting the distributions P of the image samples between both domains differ: $\mathrm{P}(X_s) \neq \mathrm{P}(X_t)$. However, the classification task is kept the same: $\mathrm{P}(Y_s|X_s) = \mathrm{P}(Y_t|X_t)$. Following the taxonomy of Wang and Deng [32], methods for domain adaptation can be categorized in discrepancy-based, adversarial-based, and reconstruction-based methods.

Discrepancy-based domain adaptation. Methods based on discrepancy use a certain criterion for fine-tuning a deep learning model to optimize it for the target domain. One type of criteria are class criteria which base the fine-tuning process on class labels [29]. Pseudo labels can be generated if no labels are available in the target domain [43]. Methods that use a statistic criterion minimize the distance between the statistical distributions of both domains, e.g., with a Kullback-Leibler divergence [46]. An architectural criterion optimizes the architecture of deep learning models to generate more domain-invariant features. Such an architectural improvement is adaptive batch normalization [21]. A geometric criterion is another type which has been used for aligning domains [4].

Adversarial-based domain adaptation. Adversarial approaches try to confuse the main network regarding the domain. They can be categorized in generative and non-generative methods. Generative methods generate a transformed input sample that contains the same content as the source sample with an appearance matching target samples [24]. Non-generative approaches reduce the domain gap in the feature space. A domain classifier is added and connected to the network via a gradient reversal layer that leads to the features being adjusted towards values most unsuitable for domain classification [9].

Reconstruction-based domain adaptation. The aim of these approaches is the reconstruction of samples from the source or target domain with the aim to achieve a domain-invariant representation of samples. The application of a combination of an encoder and a decoder with the decoder trying to reconstruct the sample from the features produced by the encoder is one approach [13]. Another approach is to use adversarial networks in the form of a Cycle-GAN that transforms a sample from one domain to the other and afterwards, reconstructs the original sample from the transformed sample [45].

4 Cross-Domain Fine-Grained Classification

In this section, an overview of existing works regarding the intersection of fine-grained classification and cross-domain classification is given. Cross-domain fine-grained classification is significantly more difficult than either of the tasks of cross-domain classification or fine-grained classification since domain adaptation and fine-grained classification are contradictive in terms of feature adjustment. While fine-grained classification requires the features to capture fine details in the image to cope with the low inter-class variance, domain adaptation is drastically changing the features in order to reduce the high inter-domain variance [33, 41].

The works are categorized by the domain adaptation setting type they apply, i.e., unsupervised, semi-supervised or supervised partially zero-shot. The different types are visualized in Figure 4.1 and an overview of the different approaches is given in Table 4.1.

Unsupervised domain adaptation. The task of unsupervised domain adaptation assumes a source domain with a large labeled dataset and a target domain with a large unlabeled dataset while both datasets include samples for all categories. The first to explore such a setting with fine-grained categories are Gebru, Hoffman, and Fei-Fei [11]. The authors exploit auxiliary attributes commonly available in fine-grained datasets by adding an additional classification head per attribute and applying an attribute consistency loss forcing the predicted attributes to match the main classification category, e.g., the body type like

Authors	Setting	Domains
Gebru et al. [11]	Unsupervised Semi-supervised	Vehicles: marketing shots, GSV
Wang et al. [33]	Unsupervised	Vehicles: marketing shots, GSV
Wang et al. [34]	Unsupervised Semi-supervised	Vehicles: marketing shots, GSV Retail products: studio images, supermarket shelves, web images
Yu, Jiang, and Li [41]	Unsupervised	Vehicles: marketing shots, GSV
Li et al. [20]	Semi-supervised	Vehicles: marketing shots, GSV
Usuyama et al. [30]	Supervised partially zero-shot	Pills: reference images, consumer images

Table 4.1: Overview of cross-domain fine-grained methods with the cross-domain setting considered and the domains between a domain adaptation was investigated. GSV stands for Google Street View.

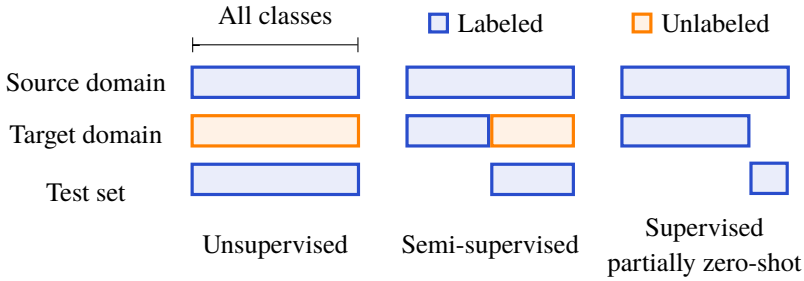


Figure 4.1: Types of settings in cross-domain fine-grained classification. The bars illustrate the classes. In an unsupervised setting, a large amount of unlabeled target samples is available for all classes. In contrast to an unsupervised setting, a part of the classes have labeled target samples available in a semi-supervised setting. The evaluation is only performed on the part of classes which have no labeled samples. In a supervised partially zero-shot setting, for a small part of the classes no target samples are available at all while this part is used for evaluation.

sedan or van matching the concrete model for vehicle classification. Since the number of training samples per attribute category is higher than per main category, attribute prediction is more stable across domains which results in a more accurate prediction of the main category due to the attribute consistency loss. The use of auxiliary attributes is additionally applied to a domain confusion loss as proposed by Tzeng et al. [29]. The approach is evaluated for vehicle classification with an adaptation from web-scraped marketing shots to Google Street View (GSV) images. Wang et al. [33] propose a quite similar approach that uses coarse-grained labels to initially train the network on an easier task that has a higher inter-class variance and is less prone to features being deteriorated during domain adaptation. The distribution of coarse-grained labels is extended to the dimension of the fine-grained labels enabling a progressive adaptation from coarse-grained to fine-grained training based on curriculum learning while using adversarial adaptation [9] to simultaneously adjust the features to the target domain. The authors also evaluate their approach on the previously mentioned vehicle classification setting with an adaptation from marketing shots to GSV images. The approach proposed by Wang et al. [34] also employs adversarial domain alignment to reduce the domain gap in the feature space. Additionally, a self-attention module is proposed that identifies class-discriminating regions and applies a part-wise classification with a result fusion step. Additional to also evaluating the marketing shots to GSV images adaptation scenario with vehicle classification, the authors propose a new setting with fine-grained classification of retail products in three domains, i.e., professional studio images, images of supermarket shelves and web images. Yu, Jiang, and Li [41] propose a method targeting fine-grained domain adaptation by maintaining quality of class-separating features during the adaptation process. This is achieved by employing domain-specific class labels with the domains being swapped after a pre-training phase resulting in domain confusion while keeping the class-separating characteristics of the features intact. Again, the vehicle classification setting adaption from marketing shots to GSV images is evaluated.

Semi-supervised domain adaptation. In the setting of semi-supervised domain adaptation, a part of the samples from the target domain are labeled. In the case of fine-grained classification, the subsets of labeled and unlabeled samples are

split by classes [11]. Gebru, Hoffman, and Fei-Fei [11] and Wang et al. [34] also evaluate their approaches explained above in a semi-supervised setting. While Gebru, Hoffman, and Fei-Fei [11] employ a cross entropy loss for the labeled samples in the target domain, Wang et al. [34] propose a contrastive loss for category-level alignment using the labeled target samples. Li et al. [20] propose the integration of a residual correction block before the final classification layer which is trained to minimize the difference between the distributions of features of the source and target domain. The decision to incorporate the residual correction block is based on the insight that early features are domain and task invariant compared to late features [38]. The authors evaluate their method in a setting adapting fine-grained vehicle classification from marketing shots to GSV images.

Supervised partially zero-shot domain adaptation. Usuyama et al. [30] are the first to propose a fine-grained domain adaptation setting different to unsupervised or semi-supervised, i.e., compared to a semi-supervised scenario, they prohibit the use of the unlabeled samples. Thus, a part of the classes has no samples available in the target domain at all making the setting more difficult. This scenario has a high practical importance since the high specificity of fine-grained classes makes it difficult to collect samples for certain classes or to ensure that a certain class is in a set of samples that has been randomly collected. The evaluation in this scenario is done on the classes that have no samples in the target domain. Such a setting is called supervised partially zero-shot following Ishii, Takenouchi, and Sugiyama [17]. Usuyama et al. [30] propose a new fine-grained dataset called ePillID containing images of pills in two domains, i.e., reference images with a specified viewpoint and lighting and a masked background and consumer images which have a greater intra-class variance. The authors evaluate a baseline approach using metric learning in a cross-domain setting.

5 Conclusion

In this work, a review of fine-grained, cross-domain, and cross-domain fine-grained classification was given. Cross-domain learning is particularly interesting for fine-grained classification due to the specificity of fine-grained classes making it highly difficult to collect abundant data for all classes. However, the challenges of fine-grained classification, i.e., a high intra-class variance and a low inter-class variance, exacerbate domain adaptation which additionally has to cope with a high inter-domain variance. Multiple approaches have been proposed to address these problems with traditional domain adaptation methods like a domain confusion loss or adversarial learning as starting point. The additional learning of auxiliary attributes or coarse-grained labels has shown to be advantageous in cross-domain scenarios and is a promising prospect for future research. The three distinguished settings for cross-domain fine-grained classification have a major impact on the applicability of the approaches. Thus, all three settings should be investigated in order to give practitioners a broad range of available approaches to solve problems with the data at hand. Particularly, the supervised partially zero-shot setting has not yet been widely explored while in practice a guarantee that images are available for all classes in the target domain is hard to provide because of the high specificity of fine-grained classes. Another interesting area for future research might be the use of synthetic data to enlarge the dataset.

References

[1] Yousef Alsahafi et al. “CarVideos: A Novel Dataset for Fine-Grained Car Classification in Videos”. In: *16th International Conference on Information Technology-New Generations (ITNG 2019)*. 2019.

[2] Marco Buzzelli and Luca Segantin. “Revisiting the CompCars Dataset for Hierarchical Car Classification: New Annotations, Experiments, and Results”. In: *Sensors* 21.2 (2021).

[3] Qianqiu Chen, Wei Liu, and Xiaoxia Yu. “A Viewpoint Aware Multi-Task Learning Framework for Fine-Grained Vehicle Recognition”. In: *IEEE Access* 8 (2020), pp. 171912–171923.

[4] Sumit Chopra, Suhrid Balakrishnan, and Raghuraman Gopalan. “Dlid: Deep learning for domain adaptation by interpolating between domains”. In: *ICML workshop on challenges in representation learning*. Vol. 2. 6. 2013.

[5] Yin Cui et al. “Large Scale Fine-Grained Categorization and Domain-Specific Transfer Learning”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2018.

[6] Jia Deng et al. “ImageNet: A large-scale hierarchical image database”. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009.

[7] Haodong Duan et al. “Omni-Sourced Webly-Supervised Learning for Video Recognition”. In: *Computer Vision – ECCV 2020*. 2020.

[8] Jie Fang et al. “Fine-Grained Vehicle Model Recognition Using A Coarse-to-Fine Convolutional Neural Network Architecture”. In: *IEEE Transactions on Intelligent Transportation Systems* 18.7 (2017), pp. 1782–1792.

[9] Yaroslav Ganin et al. “Domain-Adversarial Training of Neural Networks”. In: *Journal of Machine Learning Research* 17.59 (2016), pp. 1–35.

[10] Yang Gao et al. “Compact Bilinear Pooling”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2016.

[11] Timnit Gebru, Judy Hoffman, and Li Fei-Fei. “Fine-Grained Recognition in the Wild: A Multi-Task Domain Adaptation Approach”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Oct. 2017.

[12] Timnit Gebru et al. “Fine-Grained Car Detection for Visual Census Estimation”. In: *Proceedings of the AAAI Conference on Artificial Intelligence* 31.1 (Feb. 2017).

[13] Muhammad Ghifary et al. “Domain Generalization for Object Recognition With Multi-Task Autoencoders”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Dec. 2015.

[14] Chun-feng GUO et al. “A Survey of Fine-Grained Image Classification Based on Deep Learning”. In: *DEStech Transactions on Computer Science and Engineering* ica (2019).

[15] Shaoli Huang et al. “Part-Stacked CNN for Fine-Grained Visual Categorization”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2016.

[16] Yuqi Huo et al. “Coarse-to-Fine Grained Classification”. In: *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR’19. 2019.

[17] Masato Ishii, Takashi Takenouchi, and Masashi Sugiyama. “Partially Zero-shot Domain Adaptation from Incomplete Target Data with Missing Classes”. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. Mar. 2020.

[18] Alper Kayabasi, Kaan Karaman, and Ibrahim Batuhan Akkaya. “Comparison of distance metric learning methods against label noise for fine-grained recognition”. In: *Automatic Target Recognition XXXI*. Vol. 11729. 2021.

[19] Shu Kong and Charless Fowlkes. “Low-Rank Bilinear Pooling for Fine-Grained Classification”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. July 2017.

[20] Shuang Li et al. “Deep Residual Correction Network for Partial Domain Adaptation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43.7 (2021), pp. 2329–2344.

[21] Yanghao Li et al. “Adaptive Batch Normalization for practical domain adaptation”. In: *Pattern Recognition* 80 (2018), pp. 109–117.

[22] Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. “Bilinear CNN Models for Fine-Grained Visual Recognition”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Dec. 2015.

[23] Yen-Liang Lin et al. “Jointly Optimizing 3D Model Fitting and Fine-Grained Classification”. In: *Computer Vision – ECCV 2014*. 2014.

[24] Ming-Yu Liu and Oncel Tuzel. “Coupled Generative Adversarial Networks”. In: *Advances in Neural Information Processing Systems*. Vol. 29. 2016.

[25] Marcel Simon and Erik Rodner. “Neural Activation Constellations: Unsupervised Part Model Discovery With Convolutional Networks”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Dec. 2015.

[26] Jakub Sochor, Adam Herout, and Jiri Havel. “BoxCars: 3D Boxes as CNN Input for Improved Fine-Grained Vehicle Recognition”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2016.

[27] Kihyuk Sohn. “Improved Deep Metric Learning with Multi-class N-pair Loss Objective”. In: *Advances in Neural Information Processing Systems*. Vol. 29. 2016.

[28] Hugo Touvron et al. “Grafit: Learning Fine-Grained Image Representations With Coarse Labels”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. Oct. 2021.

[29] Eric Tzeng et al. “Simultaneous Deep Transfer Across Domains and Tasks”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Dec. 2015.

[30] Naoto Usuyama et al. “ePillID Dataset: A Low-Shot Fine-Grained Benchmark for Pill Identification”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*. June 2020.

[31] Krassimir Valev et al. “A systematic evaluation of recent deep learning architectures for fine-grained vehicle classification”. In: *Pattern Recognition and Tracking XXIX*. Vol. 10649. 2018.

[32] Mei Wang and Weihong Deng. “Deep visual domain adaptation: A survey”. In: *Neurocomputing* 312 (2018), pp. 135–153.

[33] Sinan Wang et al. “Progressive Adversarial Networks for Fine-Grained Domain Adaptation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2020.

[34] Yimu Wang et al. “An Adversarial Domain Adaptation Network for Cross-Domain Fine-Grained Recognition”. In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. Mar. 2020.

[35] Xiu-Shen Wei et al. “Fine-Grained Image Analysis with Deep Learning: A Survey”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021).

[36] Zhe Xu et al. “Webly-Supervised Fine-Grained Visual Categorization via Deep Domain Adaptation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.5 (2018), pp. 1100–1113.

[37] Hantao Yao et al. “Coarse-to-Fine Description for Fine-Grained Visual Categorization”. In: *IEEE Transactions on Image Processing* 25.10 (2016), pp. 4858–4872.

[38] Jason Yosinski et al. “How transferable are features in deep neural networks?” In: *Advances in Neural Information Processing Systems*. Vol. 27. 2014.

[39] Baosheng Yu et al. “Correcting the Triplet Selection Bias for Triplet Loss”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Sept. 2018.

[40] Chaojian Yu et al. “Hierarchical Bilinear Pooling for Fine-Grained Visual Recognition”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Sept. 2018.

[41] Han Yu, Rong Jiang, and Aiping Li. “Striking a Balance in Unsupervised Fine-Grained Domain Adaptation Using Adversarial Learning”. In: *Knowledge Science, Engineering and Management*. 2020.

[42] Lianbo Zhang et al. “Learning a Mixture of Granularity-Specific Experts for Fine-Grained Categorization”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. Oct. 2019.

[43] Xu Zhang et al. *Deep Transfer Network: Unsupervised Domain Adaptation*. arXiv:1503.00591 [cs.CV]. Mar. 2015.

[44] Chen Zhu et al. “Fine-grained Video Categorization with Redundancy Reduction Attention”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. Sept. 2018.

[45] Jun-Yan Zhu et al. “Unpaired Image-To-Image Translation Using Cycle-Consistent Adversarial Networks”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. Oct. 2017.

[46] Fuzhen Zhuang et al. “Supervised Representation Learning: Transfer Learning with Deep Autoencoders”. In: *Proceedings of the 24th International Conference on Artificial Intelligence*. IJCAI’15. 2015.

Attention Mechanism in Computer Vision: Current Status and Prospect

Chengzhi Wu

Vision and Fusion Laboratory
Institute for Anthropomatics
Karlsruhe Institute of Technology (KIT), Germany
chengzhi.wu@kit.edu

Abstract

As the key component in Transformer models, attention mechanism has shown its great power in learning feature relations even under long ranges in the natural language processing domain. Its success has also inspired researchers to apply it for computer vision tasks in recent years. In a variety of visual benchmarks, transformer-based models perform similar to or better than other types of neural networks such as convolutional and recurrent networks. In this report, we review the current status of the application of attention mechanism in computer vision tasks. In addition to categorizing the attention-based methods, since most current works are done with 2D image input and only a few focus on 3D data, we also propose research ideas in which attention mechanism is used for 3D data.

1 Introduction

Before Transformer was developed, recurrent neural networks (RNNs), e.g., GRU [7] and LSTM [16], were used in most state-of-the-art language models. However, RNNs require the information flow to be processed sequentially, which hinders the potential of parallel computation for faster sequence processing.

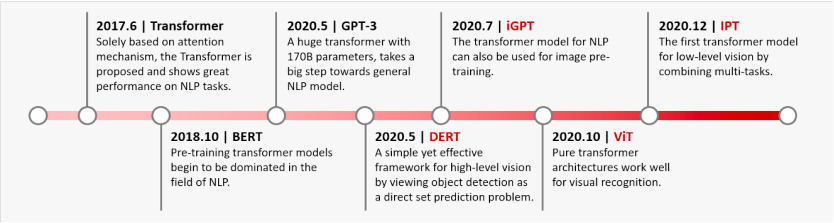


Figure 1.1: Key milestones in the development of Transformer. Models marked in red are Transformer-based models designed for CV tasks. Source from [15].

In 2017, Vaswani et al. [26] proposed Transformer, a novel encoder-decoder architecture built solely on multi-head self-attention mechanisms and feed-forward neural networks. Compared to RNNs, this attention-based model allows massive parallel computation, which enables training on larger datasets and subsequently promotes the development of large scale pre-trained models for NLP tasks. Following the pioneer work of [26], Devlin et al. [8] introduced a new language representation model named BERT to pre-train a Transformer on unlabeled text. Later, GPT-3 [1], as a massive pre-trained Transformer-based language model with 175 billion parameters, achieved astounding performance on various NLP tasks even without requiring any further fine-tuning.

On the other side, in the computer vision (CV) domain, before 2020 Convolutional neural networks (CNNs) were regarded as essential fundamentals in and almost dominate the learning methods in CV tasks. However, recent Transformer-based visual models show that they are competitive alternatives for those tasks. For image input tasks, early visual Transformer models use CNNs for first-stage latent feature computation and then apply attention methods on learned latent features, e.g. DETR [2] for image detection and DANet [12] for image segmentation. Shortly after, it is demonstrated that applying the attention mechanism solely on images is also possible for both supervised tasks, e.g. ViT [9] for classification, and self-supervised tasks, e.g. iGPT [5] for image generation. To deal with low-level vision tasks, IPT [4] combines multi-tasks in an attention-based framework and achieves great performance. Figure 1.1 shows some milestone works in the development of Transformer models, in both NLP domain and CV domain.

The key component of these Transformer models is the attention mechanism, which learns feature relations between sequence elements and even allows capturing long-term information and dependencies between them. From the feature exploitation perspective, convolution operations and attention operations are just two different ways of learning feature relations in the latent space. The former one usually focuses on local features, while the latter one usually focuses on long-range relations. There is also one interesting argument as: in our world, we discovered convolution operation before the attention operation; but probably in another parallel universe, we discovered the attention operation before the convolution operation. See more discussion in Section 3.

The rest of the report is organized as follows. Section 2 revisits the attention mechanism used in Transformers. Section 3 reviews some attention-based models for CV tasks, both on 2D data and 3D data. Since only a few works of them focus on 3D data, we propose two possible research ideas in Section 4. Finally, a conclusion is given in Section 5.

2 Revisiting the Attention Mechanism in Transformer

A Transformer consists of an encoder module and a decoder module with several encoders/decoders of the same architecture. Each encoder and decoder is composed of a self-attention layer and a feed-forward neural network. In the self-attention layer, let $X \in \mathbb{R}^{n \times d}$ denote a sequence input with n entities $(x_1, x_2, \ldots, x_n)$, where d is the embedding dimension of each entity. Multiplying by three different learnable weight matrices ($W^Q \in \mathbb{R}^{d \times d_q}$, $W^K \in \mathbb{R}^{d \times d_k}$, $W^V \in \mathbb{R}^{d \times d_v}$), each input entity x_i is first transformed into three different vectors: the query vector q_i, the key vector k_i, and the value vector v_i. They (or at least q_i and k_i) share a same dimension number, i.e., $d_q = d_k$. With a sequence of multiple entities as the input, vectors derived from different input entities are packed together into three different matrices Q, K and V. Then the output $Z \in \mathbb{R}^{n \times d_v}$ of a self-attention layer is given by:

$$Z = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{2.1}$$

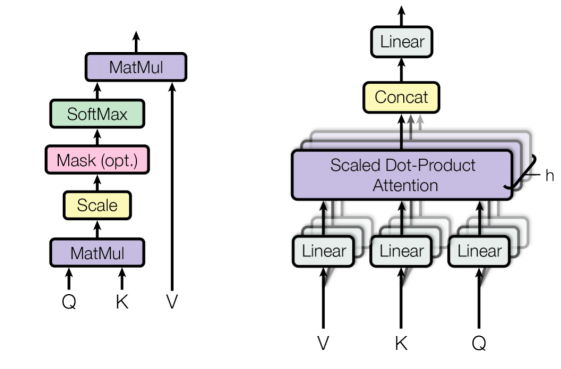
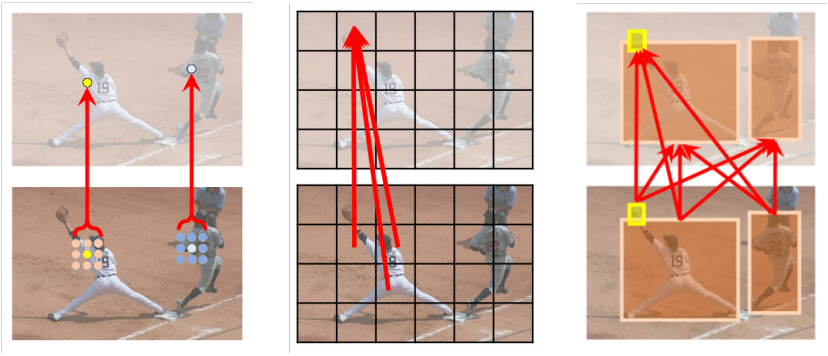


Figure 2.1: Left: self-attention. Right: multi-head attention. Source from [26].

In a more detailed explanation, Eq. 2.1 formulates the following operations. For each input entity with an embedding of vector x_i, firstly, scores between it and all other input entities are computed. Then the scores get scaled and converted into probabilities. Finally, the value vector of each input entity multiplies with the corresponding probabilities, and their sum vector z_i is the output of x_i at this layer. Additionally, similar to CNN that each convolutional layer may have multiple convolution kernels, Transformers can also use multi-head attention, as illustrated in Figure 2.1.

In recent years, researchers start to apply Transformer models on computer vision tasks, either with single-modal input or multi-modal input. Since Transformers are originally used for NLP tasks, it is quite reasonable to use them in models with both vision input and text or speech input. However, in this report, we are more interested in applying the attention mechanism on vision input, e.g. images and point clouds. Hence, models involving text or speech input will not be discussed specifically in the following sections.



(a) Pixel-wise self-attention (b) Patch-wise self-attention (c) Object-wise self-attention

Figure 3.1: Different attention types based on the different input entities from images: (a) pixel-wise self-attention; (b) patch-wise self-attention; (c) object-wise self-attention.

3 Attention Mechanism in Vision

3.1 Attention on 2D Images

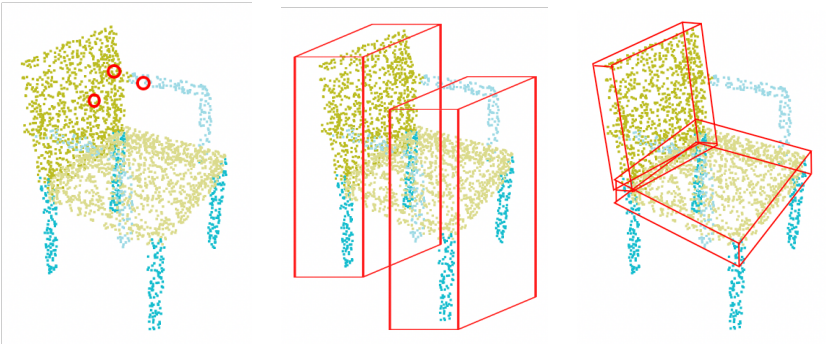
In this subsection, we review some recent papers that apply the attention-based methods on computer vision tasks. Unlike most other survey papers [15, 19] that categorize the methods based on tasks, since we are more interested in their actual application details, we categorize them based on the ways in which attentions are applied. Figure 3.1 illustrates three different attention types based on the different input entities. An interesting variant is the combination of CNNs and attention. Some models first use CNNs to get feature maps and then apply attention on learned feature maps. In this case, the feature maps can be regarded as latent representative images, but still are images.

Pixel-wise attention: iGPT [5] is a famous generative pre-trained Transformer model that learns directly on pixels. In iGPT, images are firstly downsampled and flattened into pixels, and then attention-based learning techniques that similar to GPT [1] are applied to explore autoregressive or BERT objectives. Given an input of a certain length of pixels, the model learns to predict the next

pixel; or, given an input with some pixels masked out, the model learns to predict the masked pixels. iGPT did not use any CNN, but it is also possible to combine CNNs with the attention mechanism. For example, DETR [2] does attention on flattened feature maps for image detection tasks, with positional encoding added. While DERT considers the pixel-wise attention only over the channel dimension, DANet [12] proposes to use a symmetric branch to do attention over both the channel and the position (height and depth) dimensions.

Patch-wise attention: Transformers are large models. For a high resolution image, it is too computational expensive to use every pixel as an input entity. To decrease number of input entities, apart from doing CNNs ahead, patch-wise attention is also an option. Vision Transformer (ViT) [9] is the first work to showcase how attention operations can fully replace convolution operations in deep neural networks on large-scale computer vision datasets. They applied the original Transformer model on a sequence of image patches, which are vectorized and projected to a patch embedding using a linear layer. Position embedding is attached with it to encode location information. The Transformer model was pre-trained on a large image dataset and later fine-tuned for downstream recognition benchmarks. For patch-wise attention, it is also possible to combine it with CNNs. For example, after getting the feature maps, IPT [4] groups patches along the channel dimension for attention computation.

Object-wise attention: Different from above two types, this type of attention models does not directly learn from the original images. It is more of a second process in the whole learning pipeline and requires some additional information output from the previous process. For example, based on rough detection results, Relation Networks [17] processes a set of objects simultaneously through interaction between their appearance feature and geometry, thus allowing modeling of their relations. Moreover, when the time dimension is involved, doing object-wise attention also enhance the performance for tasks like action recognition and object tracking in videos. Interestingly, when input entities can be formalized as graphs, attentions can be applied not only in the fashion of key-query attention. For example, in the work of [27] which does action recognition on videos, graph-based attention is applied to learn object-wise relations. See more relevant discussions on graph-based attention in subsection 3.2.



(a) Point-wise self-attention (b) Patch-wise self-attention (c) Part-wise self-attention

Figure 3.2: Different attention types based on the different input entities from point clouds: (a) point-wise self-attention, attention is applied over all points or only neighbor points; (b) patch-wise self-attention, special subsample operations may be developed for patch choices; (c) part-wise self-attention, points of semantic parts or 3D bounding boxes may be used as input entities, in scene point clouds it would be object-wise attention.

3.2 Attention on 3D Data

Apart from RGBD images and multi-view images, other widely used 3D data representations are volumetric data, point clouds, meshes, etc. For voxel-based data representation, except for the methods on point clouds that voxelize the whole point cloud space [23, 29], there is really very little work that applies attention on volumetric data directly. We only find one work that does volumetric spatial and channel attention on medical images for segmentation and detection [28], but the method still treats input as image-based structure, other than in 3D. For mesh data representation, there are also only a few works that apply attention on them, e.g. METRO [22] uses Transformer models for human pose and mesh reconstruction. We believe there will be more explorations in applying attention on those data representations in the following years.

Of all the 3D data representations, point clouds are mostly investigated since they are more often used in the real-world applications, e.g. autonomous driving or industrial inspection. Hence, in the following part of this subsection, we

mainly review the attention-based models for point clouds. Figure 3.2 illustrates three different attention types based on the different input entities from point clouds. Applying attention on point clouds is not the same as applying attention on images. Unlike images that are composed of well-aligned pixels, points in a point cloud are usually unordered and randomly positioned. This has pros and cons. On one side, it requires no positional encoding since we want to rule out the influence of point order. On the other side, having an unfixed number of neighbor points makes the common convolution operations unfeasible. In short, point clouds do not have image-like feature maps, thus the combination of convolution and attention operations becomes difficult.

Point-wise attention: PCT [14] pioneered on this topic by replacing the encoder in the original PointNet [24] framework with some attention-based layers. Skip links are also used for better latent feature acquisition. In [30], the point Transformer layer is sandwiched between two linear layers to create a point Transformer block, which is stacked multiple times in the proposed network architecture. Their design also includes transition down/up blocks to reduce/increase the number of points between consecutive layers, in a typical encoding-decoding style.

Different from above methods that use key-query attention for computing the attention score, there are also other methods use MLP and softmax layers directly to compute the attention score. RandLA-Net [18] learns attention scores for points as a soft mask to replace the original max/mean/sum pooling layer for better feature pooling. GAPNet [3] and Liang et al. [21] do similar point-wise self-attention with neighbor points to learn attention coefficients from them for each point. We term this type of attention as aforementioned graph-based attention, since it is usually applied on graph structure data (finding neighbor relations for each point is like building edges that defined in graphs).

Patch-wise attention: In order to perform parallel computation, if patches are not processed into fixed-length latent representation firstly, they need to be of the same size as input entities. Unlike images which are easy to cut into same size of patches, how to subsample the point clouds into patches is still an open question. Engel et al. [11] proposes an interesting idea of SortNet, with which sub-point clouds of same size can be learned. After that, the attention operation is applied on latent features of sub-point clouds and the global feature to perform

local-global attention. However, the patches learned with SortNet are only of little semantic meanings. Even with multiple-SortNets, only a small ratio of points are selected to represent the input shape. Improvements may be made on subsample operations to get more representative patches.

Part-wise attention: This type of attention learns relations between semantic areas in a point cloud. For shape point clouds, it is part-wise self-attention. For scene point clouds, it is object-wise self-attention. Same as in applying object-wise attention on images, here it also requires a former step of getting some additional information. For example, MPAN [20] firstly does segmentation on the input shape point cloud, then do part-wise self-attention over segmented parts to get a shape descriptor for shape retrieval tasks.

4 Research Ideas for Attention with 3D Data

As discussed above, only a few researchers work on applying the attention mechanism directly on 3D Non-Euclidean data. Making some possible progresses on this topic would be exciting. In this section, two research ideas are proposed in this domain. One for voxel-based data, the other for point cloud data.

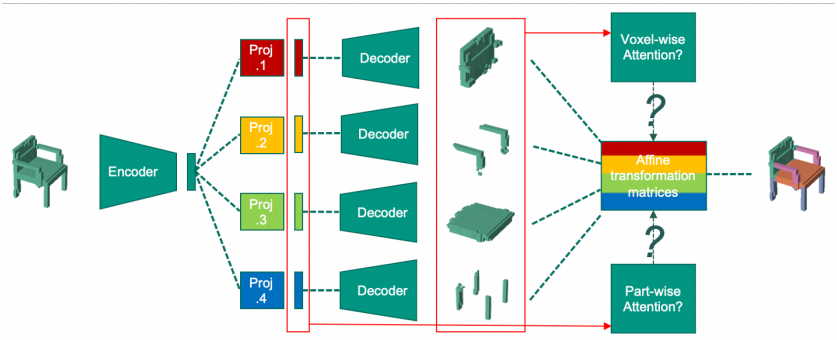


Figure 4.1: Proposed pipeline of the VoxAttention framework.

4.1 VoxAttention: Volumetric Shape Synthesis via Part Assembly

As discussed in section 3.2, apart from the methods that perform voxelization on point clouds, works that directly apply the attention mechanism on 3D volumetric data are really rare. We think it would be an interesting topic to research on. Among all the voxel data-based tasks, deep generative models-based 3D shape synthesis is getting more attention recently since they are not only good at data reconstruction, but also helpful in producing meaningful latent representations. Current existing 3D shape synthesis methods can be divided into two kinds: structure-oblivious ones and structure-aware ones. Compared to structure-oblivious methods, structure-aware methods introduce additional semantic information and thus result in better performances. They are starting to gain more and more attention in this domain. Schor et al. [25] train part-wise generators and a part composition network for the generation of 3D point clouds. Dubrovina et al. [10] propose a decomposer-composer framework to learn a factorized shape embedding space for part-based 3D volumetric shape modeling.

Inspired by the work of [10], we propose a new generator-assembler framework for 3D volumetric shape synthesis, with the additional help of self-attention mechanism. Figure 4.1 illustrates its basic idea. Firstly, the binary voxel grid is fed into an encoder, resulting in a latent encoding. Then several projection matrices are used to project it to different part latent representations, and a parameter-shared decoder is used to reconstruct the parts. To disentangle the semantic information as much as possible, the projection matrices are mutual orthogonal. Note the reconstructed parts now are enlarged and centered in a 32^3 voxel space. To learn respective affine transformation matrices for part assembly, instead of using a straightforward idea to apply CNNs on the reconstructed parts like [10], our proposal is to use the attention mechanism. For example, we can apply part-wise attention over part latent representations, or do voxel-wise attention on reconstructed parts to learn spatial relations between them. With our proposed attention-based method, ideally, the network should be able to learn dynamic transformation matrices for part latent representations. Even if some part latent representations are swapped, the network should still be able to reconstruct shapes with relatively correct part size and position.

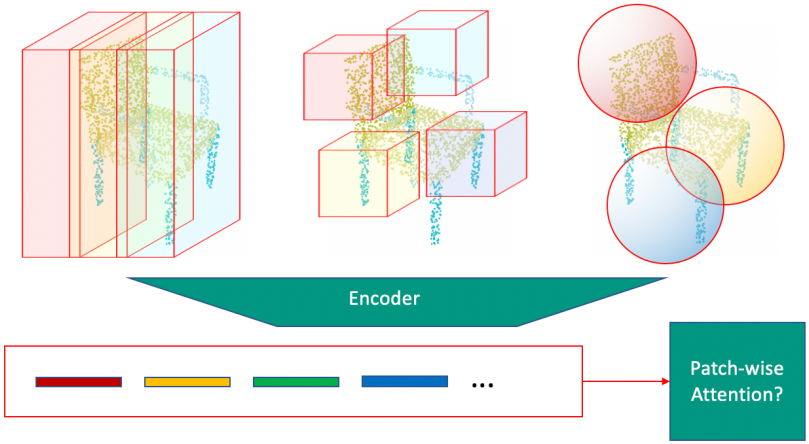


Figure 4.2: Proposed pipeline of the PointAttention framework.

4.2 PointAttention: Patch-wise Attention-based Learning on Point Clouds

Both key-query attention and graph-based attention are more often used for the point cloud-related tasks. However, most of these works only focus on point-wise self-attention. We propose to explore more on patch-wise self-attention. As discussed in subsection 3.2, how to subsample the point clouds into patches is still an open question. Unlike [23] or [29] in which the point cloud space get voxelized, we directly use pre-designed subsampling methods to cut out patches, as illustrated on the top of Figure 4.2.

A key point is to normalize either the input or the latent feature to the same size. In [23, 29], although each voxel contains different number of points, voxels' feature maps or latent representations are of same size. In our case, we can directly subsample the original point clouds into same size of sub-point clouds and use them as the network input instead. Apart from the direct FPS (short for farthest point sampling) subsample operation, we propose such a new one: firstly slice the point cloud or use a cuboid/sphere to cut the point cloud, then apply FPS

on these sub-point clouds to get patches of a required point number. Afterwards, patch-wise attention may be applied on latent representations encoded from different patches to learn patch relations for better point cloud classification and segmentation. Another idea is that instead of pre-defining the subsample operations manually, we can define a module similar to SortNet [11] to let it learn better patches by itself automatically.

If the original point clouds are not of large size, the subsamples can still somehow represent the 3D shape partially with semantic meanings. In this case, the subsampling operations are augmentation-like operations, hence the subsamples are also ideal input data for contrastive learning [6, 13]. Combining the attention mechanism and contrastive learning, other new self-supervised learning methods may also be proposed.

5 Conclusion

Attention mechanism plays an important role in learning feature relations between input entities, especially for the long-range relations. In this report, we review a variety of attention-based approaches that are used in the NLP or the CV domain. In the CV domain, based on the type of the input entities, we categorize the attention-based methods into different types for 2D data and 3D data respectively. We also share some insights from this perspective. Finally, since applying attention on 3D data is relatively more difficult and there exists only a few works, we propose two possible research ideas in this scope. Future researches will firstly take the proposed ideas into consideration and perform experiments on them. We also hope this technical report may inspire other researchers that are interested in this topic.

References

[1] Tom B. Brown et al. “Language Models are Few-Shot Learners”. In: *ArXiv* abs/2005.14165 (2020).

[2] Nicolas Carion et al. “End-to-End Object Detection with Transformers”. In: *ArXiv* abs/2005.12872 (2020).

[3] Can Chen, Luca Zanotti Fragonara, and Antonios Tsourdos. “GAPNet: Graph Attention based Point Neural Network for Exploiting Local Feature of Point Cloud”. In: *Neurocomputing* 438 (2021), pp. 122–132.

[4] Hanting Chen et al. “Pre-Trained Image Processing Transformer”. In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021), pp. 12294–12305.

[5] Mark Chen et al. “Generative Pretraining From Pixels”. In: *ICML*. 2020.

[6] Ting Chen et al. “A Simple Framework for Contrastive Learning of Visual Representations”. In: *ArXiv* abs/2002.05709 (2020).

[7] Junyoung Chung et al. “Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling”. In: *ArXiv* abs/1412.3555 (2014).

[8] Jacob Devlin et al. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. In: *NAACL*. 2019.

[9] Alexey Dosovitskiy et al. “An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale”. In: *ArXiv* abs/2010.11929 (2021).

[10] Anastasia Dubrovina et al. “Composite Shape Modeling via Latent Space Factorization”. In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (2019), pp. 8139–8148.

[11] Nico Engel, Vasileios Belagiannis, and Klaus C. J. Dietmayer. “Point Transformer”. In: *IEEE Access* 9 (2021), pp. 134826–134840.

[12] J. Fu et al. “Dual Attention Network for Scene Segmentation”. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2019), pp. 3141–3149.

[13] Jean-Bastien Grill et al. “Bootstrap Your Own Latent: A New Approach to Self-Supervised Learning”. In: *ArXiv* abs/2006.07733 (2020).

[14] Meng-Hao Guo et al. “PCT: Point Cloud Transformer”. In: *Comput. Vis. Media* 7 (2021), pp. 187–199.

[15] Kai Han et al. “A Survey on Visual Transformer”. In: *ArXiv* abs/2012.12556 (2020).

[16] Sepp Hochreiter and Jürgen Schmidhuber. “Long Short-Term Memory”. In: *Neural Computation* 9 (1997), pp. 1735–1780.

[17] Han Hu et al. “Relation Networks for Object Detection”. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018), pp. 3588–3597.

[18] Qingyong Hu et al. “RandLA-Net: Efficient Semantic Segmentation of Large-Scale Point Clouds”. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020), pp. 11105–11114.

[19] Salman Hameed Khan et al. “Transformers in Vision: A Survey”. In: *ArXiv* abs/2101.01169 (2021).

[20] Ziru Li et al. “MPAN: Multi-Part Attention Network for Point Cloud Based 3D Shape Retrieval”. In: *IEEE Access* 8 (2020), pp. 157322–157332.

[21] Zhidong Liang et al. “3D Instance Embedding Learning With a Structure-Aware Loss Function for Point Cloud Segmentation”. In: *IEEE Robotics and Automation Letters* 5 (2020), pp. 4915–4922.

[22] Kevin Lin, Lijuan Wang, and Zicheng Liu. “End-to-End Human Pose and Mesh Reconstruction with Transformers”. In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021), pp. 1954–1963.

[23] Jiageng Mao et al. “Voxel Transformer for 3D Object Detection”. In: *ArXiv* abs/2109.02497 (2021).

[24] C. Qi et al. “PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), pp. 77–85.

[25] Nadav Schor et al. “CompoNet: Learning to Generate the Unseen by Part Synthesis and Composition”. In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (2019), pp. 8758–8767.

[26] Ashish Vaswani et al. “Attention is All you Need”. In: *ArXiv* abs/1706.03762 (2017).

[27] X. Wang and Abhinav Gupta. “Videos as Space-Time Region Graphs”. In: *ECCV*. 2018.

[28] Xudong Wang et al. “Volumetric Attention for 3D Medical Image Segmentation and Detection”. In: *MICCAI*. 2019.

[29] Cheng Zhang et al. “PVT: Point-Voxel Transformer for 3D Deep Learning”. In: 2021.

[30] Hengshuang Zhao et al. “Point Transformer”. In: *ArXiv* abs/2012.09164 (2020).

Karlsruher Schriftenreihe zur Anthropomatik (ISSN 1863-6489)

Band 1 Jürgen Geisler
Leistung des Menschen am Bildschirmarbeitsplatz.
ISBN 3-86644-070-7

Band 2 Elisabeth Peinsipp-Byma
Leistungserhöhung durch Assistenz in interaktiven Systemen zur Szenenanalyse. 2007
ISBN 978-3-86644-149-1

Band 3 Jürgen Geisler, Jürgen Beyerer (Hrsg.)
Mensch-Maschine-Systeme.
ISBN 978-3-86644-457-7

Band 4 Jürgen Beyerer, Marco Huber (Hrsg.)
Proceedings of the 2009 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory.
ISBN 978-3-86644-469-0

Band 5 Thomas Usländer
Service-oriented design of environmental information systems.
ISBN 978-3-86644-499-7

Band 6 Giulio Milighetti
Multisensorielle diskret-kontinuierliche Überwachung und Regelung humanoider Roboter.
ISBN 978-3-86644-568-0

Band 7 Jürgen Beyerer, Marco Huber (Hrsg.)
Proceedings of the 2010 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory.
ISBN 978-3-86644-609-0

Band 8 Eduardo Monari
Dynamische Sensorselektion zur auftragsorientierten Objektverfolgung in Kameranetzwerken.
ISBN 978-3-86644-729-5

Die Bände sind unter www.ksp.kit.edu als PDF frei verfügbar oder als Druckausgabe bestellbar.

Band 9 Thomas Bader
Multimodale Interaktion in Multi-Display-Umgebungen.
ISBN 3-86644-760-8

Band 10 Christian Frese
Planung kooperativer Fahrmanöver für kognitive Automobile.
ISBN 978-3-86644-798-1

Band 11 Jürgen Beyerer, Alexey Pak (Hrsg.)
Proceedings of the 2011 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory.
ISBN 978-3-86644-855-1

Band 12 Miriam Schleipen
Adaptivität und Interoperabilität von Manufacturing Execution Systemen (MES).
ISBN 978-3-86644-955-8

Band 13 Jürgen Beyerer, Alexey Pak (Hrsg.)
Proceedings of the 2012 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory.
ISBN 978-3-86644-988-6

Band 14 Hauke-Hendrik Vagts
Privatheit und Datenschutz in der intelligenten Überwachung: Ein datenschutzgewährendes System, entworfen nach dem „Privacy by Design" Prinzip.
ISBN 978-3-7315-0041-4

Band 15 Christian Kühnert
Data-driven Methods for Fault Localization in Process Technology. 2013
ISBN 978-3-7315-0098-8

Band 16 Alexander Bauer
Probabilistische Szenenmodelle für die Luftbildauswertung.
ISBN 978-3-7315-0167-1

Band 17 Jürgen Beyerer, Alexey Pak (Hrsg.)
Proceedings of the 2013 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory.
ISBN 978-3-7315-0212-8

Die Bände sind unter www.ksp.kit.edu als PDF frei verfügbar oder als Druckausgabe bestellbar.

Band 18 Michael Teutsch
Moving Object Detection and Segmentation for Remote Aerial Video Surveillance.
ISBN 978-3-7315-0320-0

Band 19 Marco Huber
Nonlinear Gaussian Filtering: Theory, Algorithms, and Applications.
ISBN 978-3-7315-0338-5

Band 20 Jürgen Beyerer, Alexey Pak (Hrsg.)
Proceedings of the 2014 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory.
ISBN 978-3-7315-0401-6

Band 21 Todor Dimitrov
Permanente Optimierung dynamischer Probleme der Fertigungssteuerung unter Einbeziehung von Benutzerinteraktionen.
ISBN 978-3-7315-0426-9

Band 22 Benjamin Kühn
Interessengetriebene audiovisuelle Szenenexploration.
ISBN 978-3-7315-0457-3

Band 23 Yvonne Fischer
Wissensbasierte probabilistische Modellierung für die Situationsanalyse am Beispiel der maritimen Überwachung.
ISBN 978-3-7315-0460-3

Band 24 Jürgen Beyerer, Alexey Pak (Hrsg.)
Proceedings of the 2015 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory.
ISBN 978-3-7315-0519-8

Band 25 Pascal Birnstill
Privacy-Respecting Smart Video Surveillance Based on Usage Control Enforcement.
ISBN 978-3-7315-0538-9

Band 26 Philipp Woock
Umgebungskartenschätzung aus Sidescan-Sonardaten für ein autonomes Unterwasserfahrzeug.
ISBN 978-3-7315-0541-9

Band 27 Janko Petereit
Adaptive State × Time Lattices: A Contribution to Mobile Robot Motion Planning in Unstructured Dynamic Environments.
ISBN 978-3-7315-0580-8

Band 28 Erik Ludwig Krempel
Steigerung der Akzeptanz von intelligenter Videoüberwachung in öffentlichen Räumen.
ISBN 978-3-7315-0598-3

Band 29 Jürgen Moßgraber
Ein Rahmenwerk für die Architektur von Frühwarnsystemen. 2017
ISBN 978-3-7315-0638-6

Band 30 Andrey Belkin
World Modeling for Intelligent Autonomous Systems.
ISBN 978-3-7315-0641-6

Band 31 Chettapong Janya-Anurak
Framework for Analysis and Identification of Nonlinear Distributed Parameter Systems using Bayesian Uncertainty Quantification based on Generalized Polynomial Chaos.
ISBN 978-3-7315-0642-3

Band 32 David Münch
Begriffliche Situationsanalyse aus Videodaten bei unvollständiger und fehlerhafter Information.
ISBN 978-3-7315-0644-7

Band 33 Jürgen Beyerer, Alexey Pak (Eds.)
Proceedings of the 2016 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory.
ISBN 978-3-7315-0678-2

Band 34 Jürgen Beyerer, Alexey Pak and Miro Taphanel (Eds.)
Proceedings of the 2017 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory.
ISBN 978-3-7315-0779-6

Band 35 Michael Grinberg
Feature-Based Probabilistic Data Association for Video-Based Multi-Object Tracking.
ISBN 978-3-7315-0781-9

Band 36 Christian Herrmann
Video-to-Video Face Recognition for Low-Quality Surveillance Data.
ISBN 978-3-7315-0799-4

Band 37 Chengchao Qu
Facial Texture Super-Resolution by Fitting 3D Face Models.
ISBN 978-3-7315-0828-1

Band 38 Miriam Ruf
Geometrie und Topologie von Trajektorienoptimierung für vollautomatisches Fahren.
ISBN 978-3-7315-0832-8

Band 39 Angelika Zube
Bewegungsregelung mobiler Manipulatoren für die Mensch-Roboter-Interaktion mittels kartesischer modellprädiktiver Regelung.
ISBN 978-3-7315-0855-7

Band 40 Jürgen Beyerer and Miro Taphanel (Eds.)
Proceedings of the 2018 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory.
ISBN 978-3-7315-0936-3

Band 41 Marco Thomas Gewohn
Ein methodischer Beitrag zur hybriden Regelung der Produktionsqualität in der Fahrzeugmontage.
ISBN 978-3-7315-0893-9

Band 42 Tianyi Guan
Predictive energy-efficient motion trajectory optimization of electric vehicles.
ISBN 978-3-7315-0978-3

Band 43 Jürgen Metzler
Robuste Detektion, Verfolgung und Wiedererkennung von Personen in Videodaten mit niedriger Auflösung.
ISBN 978-3-7315-0968-4

Band 44 Sebastian Bullinger
Image-Based 3D Reconstruction of Dynamic Objects Using Instance-Aware Multibody Structure from Motion.
ISBN 978-3-7315-1012-3

Band 45 Jürgen Beyerer, Tim Zander (Eds.)
Proceedings of the 2019 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory.
ISBN 978-3-7315-1028-4

Band 46 Stefan Becker
Dynamic Switching State Systems for Visual Tracking.
ISBN 978-3-7315-1038-3

Band 47 Jennifer Sander
Ansätze zur lokalen Bayes'schen Fusion von Informationsbeiträgen heterogener Quellen.
ISBN 978-3-7315-1062-8

Band 48 Philipp Christoph Sebastian Bier
Umsetzung des datenschutzrechtlichen Auskunftsanspruchs auf Grundlage von Usage-Control und Data-Provenance-Technologien.
ISBN 978-3-7315-1082-6

Band 49 Thomas Emter
Integrierte Multi-Sensor-Fusion für die simultane Lokalisierung und Kartenerstellung für mobile Robotersysteme.
ISBN 978-3-7315-1074-1

Band 50 Patrick Dunau
Tracking von Menschen und menschlichen Zuständen.
ISBN 978-3-7315-1086-4

Band 51 Jürgen Beyerer, Tim Zander (Eds.)
Proceedings of the 2020 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory.
ISBN 978-3-7315-1091-8

Band 52 Lars Wilko Sommer
Deep Learning based Vehicle Detection in Aerial Imagery.
ISBN 978-3-7315-1113-7

Band 53 Jan Hendrik Hammer
Interaktionstechniken für mobile Augmented-Reality-Anwendungen basierend auf Blick- und Handbewegungen.
ISBN 978-3-7315-1169-4

Band 54 Jürgen Beyerer, Tim Zander (Eds.)
Proceedings of the 2021 Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory.
ISBN 978-3-7315-1171-7

Lehrstuhl für Interaktive Echtzeitsysteme
Karlsruher Institut für Technologie

Fraunhofer-Institut für Optronik, Systemtechnik
und Bildauswertung IOSB Karlsruhe

In 2021, the annual joint workshop of the Fraunhofer Institute of Optronics, System Technologies and Image Exploitation (IOSB) and the Vision and Fusion Laboratory (IES) of the Institute for Anthropomatics, Karlsruhe Institute of Technology (KIT) was hosted at the IOSB in Karlsruhe for the first time due to the pandemic and the strict regulations of such an event in this times. For a week from the 2nd to the 6th July the doctoral students of both institutions presented extensive reports on the status of their research and discussed topics ranging from computer vision and optical metrology to network security, usage control and machine learning. The results and ideas presented at the workshop are collected in this book in the form of detailed technical reports. This volume provides a comprehensive and up-to-date overview of the research program of the IES Laboratory and the Fraunhofer IOSB.

ISSN 1863-6489 (Schriftenreihe)
ISSN 2510-7259 (Tagungsband)
ISBN 978-3-7315-1171-7