OSTBAYERISCHE
TECHNISCHE HOCHSCHULE
REGENSBURG

Integration of Optical Computation in Processor Architectures for Automotive Use Cases.

Bachelor thesis

Department of Computer Science and Mathematics of the
Ostbayerische Technische Hochschule Regensburg
in the degree program
Computer Science

submitted in partial Fulfillment of the Requirement

to obtain the academic degree of
Bachelor of Science (B.Sc.)

Submitted by:	Florian Schulenberg
Matriculation number:	3271221
First reviewer:	Prof. Dr. rer nat. Wolfgang Mauerer
Second reviewer:	Prof. Dr. Johannes Schildgen
Supervisor:	Dr. Ralf Ramsauer
Submission date:	1. March 2024

Contents

List of Figures

List of Tables

Abbreviations

5GAA	5G Automotive Association
ABS	Antilock Braking System
ADC	Analogue Digital Converter
ADAS	Advanced Driver-Assistance Systems
ALU	Arithmetic Logic Unit
AMBA	ARM's Advanced Microcontroller Bus Architecture
AR	Augmented Reality
ASIC	application-specific integrated circuit
ASIL	Automotive Safety Integrity Level
BEV	Battery Electric Vehicle
BMW	Bayerische Motoren Werke
CEO	Chief Executive Officer
CES	Consumer Electronics Show
CID	Central Information Display
CISC	Complex Instruction Set Computing
CNN	Convolutional Neural Networks
DAC	Digital Analogue Converter
DMA	direct memory access
DSC	Dynamic Stability Control
ESA	European Space Agency
FLAC	free lossless Audio Codec
FPGA	Field-programmable Gate Array
FPS	Frames per Second
FSD	Full Self Driving
FTTH	Fibre to the Home
GLSL	OpenGL Shading Language
HBM	High Bandwidth Memory
I2C	Inter-Integrated Circuit
IR	Infra-Red
ISA	Instruction Set Architecture
KPI	Key Performance Indicator
LCI	Life Cycle Impulse
LIDAR	Light detection and ranging
MDM	Mode division multiplexing
MPLC	Multiplane Light Conversion
MRR	microring resonator
MXU	matrix-multiply unit
MZI	Mach Zehnder Interferometer

MVM	matrix vector multiplication
NPU	neuromorphic processing unit
NN	Neural Network
NoC	Network on Chip
ODD	Operational Design Domain
OEM	Original Equipment Manufacturer
OPU	Optical Processing Unit
PCIe	Peripheral Component Interconnect Express
PLC	Plane Lightwave Conversion
PPS	points per second
PU	processing unit
RISC	Reduced Instruction Set Computing
SLAM	Simultaneous Localization and Mapping
SLM	spatial light modulators
SLR	Service Level Requirement
SoC	System on Chip
SPI	Serial Peripheral Interface
TOPS	Tera operations per second
TPU	Tensor Processing Unit
V2X	Vehicle-to-Anything Communication
VR	Virtual Reality
WDM	Wavelength Division Multiplexing
XR	Extended Reality
YOLO	You Only Look Once

1 Introduction

This thesis is based on a research topic within the BMW Group research department.

1.1 The Landscape In Automotive

The automotive industry is within a major transformation and a shift of values. After vehicles would mainly differentiate themselves by their performance and driving characteristics, digital services and features are now playing a major role in customers purchasing decisions. Whilst range and performance of a specific vehicle is still perceived as important with the selection of a Battery Electric Vehicle (BEV), the digital experience plays a key role for customer satisfaction. This experience consists of a variety of digital features, that range from the integration of personal devices, intelligent navigation, in-car entertainment and gaming to an integration of the vehicle into a greater ecosystem [37], [157]. The Bayerische Motoren Werke (BMW) Group is positioning itself as a manufacturer of premium mobility and is currently serving 20.2 Million connected vehicles in 75 global markets, in which the customer journey is greatly dependent on offered digital services and features of the vehicle [28]. Aside mentioned in-car services, increasing levels of vehicle autonomy (ADAS), that is hoped to revolutionize the idea individual mobility (like autonomous taxis like Volt in San Francisco) or at least make a drivers journey more comfortable [52].

The addition of digital experiences, automated/assisted driving and ecosystem integration further increase requirements on available computational power within the vehicle and connectivity to the outside world [17]. The interest in Advanced Driver-Assistance Systems (ADAS) especially is fostering this need, as understanding the surroundings of the vehicle requires a great amount of sensing and data processing [265]. Adding additional computing power within automotive-embedded system is no simple feat. It's more complicated because of issues like limited space, energy consumption and the heat generated. These problems mean that increasing power isn't straightforward and requires careful thinking and consideration for balancing processing-power needed against energy-consumption and interior-design.

This need of greater computational offerings is long present within the automotive industry, just like it is within the consumer and cross-industry landscape. The Original Equipment Manufacturer (OEM) industry is used to create an invitation of tenders for new vehicle generations, where computational demand is going along with offerings by traditional chip-developers (as far as known to public). Relatively new however is the rapid increase of computation in relative numbers thanks to the use of machine learning within the vehicle [17] and the evolving concepts in computation, that are being used to tackle an increasing demand for efficient computation. This trend can be seen within Tesla's own custom chip design, where two neuromorphic processing unit (NPU)'s are used to accelerate machine learning use cases [254]. Another example is the Mercedes EQXX concept car that uses an NPU for speech detection [277]. These examples show the great interest of automotive in developing or integrating computational hardware, that is highly efficient for a specific use case, like low-energy accelerators for AI use cases or signal processing [254], [277].

1.2 Trends in Computation

Ever since the first processor was introduced in the 1970s computational power as been a major enabler for innovation across all industries [77]. The development of processors has become a crucial factor in the advancement of both technology and society. The overall goal since then remains faster processing, greater throughput and energy efficiency. Engineers have since then use several concepts to enhance processors.

One of the key-changes on processors throughout the year is the size of hardware. The Intel Pentium III Coppermine processor introduced in 1999, was manufactured on a newly introduced 180 nm node [112] for instance. The successive Intel Pentium IV has featured a smaller 90 nm node just 5 years later. The year of 2023 features various processors manufactured on a 3 nm node including the current generation of Apple Mac and Phone processors [154]. Production of 2 nm nodes are currently on the way with a market entry later in the 2020s [58]. A smaller node offers for more transistors on the same area / die size, thus offering space to more Arithmetic Logic Unit (ALU) or caches for increasing computational throughput per clock-cycle. The increase of transistors per cheap is approximated through Moore's Law being an empirical observation that the number of transistors in an integrated circuit doubles approximately every two years. It was first described by Gordon E. Moore, the co founder of Intel, in 1965. Moore's Law is not a law of nature, but rather a long-term trend in how technology is changing. The observation has held true for more than half a century, driving exponential growth in the number of transistors on integrated circuits and leading to significant advancements in computing power and capabilities [167]. It is greatly discussed, when and if Moore's Law will end. Past years brought up several papers discussing, if a current node would be the commercially smallest achievable node [282], yet even smaller nodes are now in mass market production [154]. Current research suggests, that a physical minimum gate length is at 0,34 nm; the size of a single carbon atom [208]. With challenges in both commercialisation and physical feasibility it is currently unknown, how much smaller transistors can become [191].

In contrast to the development of processing power stands the increasing demand for digital products, the increasing amount of data to process and the rising demand of such. Increasing usage of and application of machine learning is a great example for the rising demand for processing power. The usage of AI of industry and consumer markets is greatly increasing, while the effort required to train and execute underlying Neural Network (NN) is increasing exponentially [212], [222]. While increasingly well performing processors and GPU are countering increasing demand, an increasing energy consumption is troublesome for battery run devices (including battery electric vehicles for instance).

The evolution of computational capabilities stands in stark contrast to the burgeoning demand for digital products, the exponential growth in data volumes, and the escalating requirements for data processing. A quintessential illustration of this growing demand for computational resources is the pervasive adoption and integration of machine learning technologies. The application of artificial intelligence (AI) within both industrial and consumer sectors is witnessing a significant surge, paralleling an exponential increase in the computational effort required to train and implement underlying Neural Network [212], [222].

As processors and graphics processing units (GPUs) evolve to offer superior performance, they serve as a counterbalance to the increasing demands for processing power. However, this advancement is accompanied by a notable increase in energy consumption, presenting a significant challenge for devices reliant on battery power. This issue is especially troublesome in the context of battery electric vehicles (BEVs) and other portable electronic devices, where energy efficiency is paramount [245]. The trade-off between enhancing computational performance and managing energy consumption necessitates a careful consideration of the implications for sustainability and device autonomy.

Specialised hardware is found to be a chance to enable more sustainable growth of increasing computational performance. The introduction and usage of optical components is considered to be one option to achieve just that [159], [278], [302]. Interest into the use of photonics is greatly increasing [156], [164], [260] and hopes are, that optical accelerators could increase energy efficiency and performance greatly.

The ongoing research and introduction of specialised hardware [15], [218], [254] presents a promising approach for enabling sustainable growth in computational performance. New concepts like neuromorphic computation [61], biological computation (*e.g.* DNA compute) [76] or optical computation are envisioned to accelerate processing power. The integration and application of optical components specifically emerge as a possibility in this context, with high hopes in its ability to increase computational power and efficiency [159], [278], [302]. The growth in funding and investments on optical technology and its startups [156], [164], [260] is indicative of the optimism surrounding optical accelerators, which are anticipated to significantly enhance both energy efficiency and computational performance. This shift towards optical technology not only aims to meet growing demands for processing capabilities but also aligns with the imperative for energy conservation and environmental sustainability.

1.3 Objective and Conclusion

The growing proliferation of digital services and products within automotive environments, coupled with the increasing complexity and energy demands of achieving higher computational performance, renders the integration of hardware accelerators into the vehicular landscape both compelling and opportune. In this thesis, I aim to provide a comprehensive overview of the developments, potentials, and challenges associated with the theoretical implementation of an OPU within vehicular systems. I will further explore the physical principles underlying photonics and its application in data processing, review potential use cases within automotive contexts, and offer insights into trends and advancements in processor technology. An examination of vehicle services will highlight areas where optical computation could facilitate enhanced processing efficiency, followed by a detailed analysis of the algorithms and methodologies employed, to deepen understanding of the specific requirements involved. Through a synthesis of knowledge on photonic hardware acceleration and vehicular applications, this work will present a high-level architectural proposal that delineates the feasible applications and constraints of an OPU. The overarching objective of this thesis is to investigate whether photonic processing units represent a viable option for achieving increased energy efficiency and improved performance in vehicles of the future, while also addressing potential obstacles encountered in this endeavor.

2 State of Research

This Section provides a comprehensive overview of the three principal motivations underpinning the topic of optical computation for vehicle use cases. First examined is the burgeoning interest and ongoing research into the domain of optical/photonic computation, as done so by startup Lightmatter [278] and [67]. Within this context, I will conduct a comparative analysis of four distinct methodologies for realising computation through propagation of light. Additionally, I explore the current trends and concerted efforts aimed at enhancing computational capabilities. Subsequently, it will assess the processing demands encountered within the context of a moving vehicle for products of the next years. This evaluation aims to identify use cases that are presently underserved or are anticipated to encounter performance bottlenecks. This inquiry not only underscores the critical need for advancements in computational strategies, but also highlights the potential application areas that could benefit from photonic acceleration.

2.1 Optical Computation

In the quest for computational advancements, photonics for the use within computing stands out as an innovation alongside other novel concepts like neuromorphic computing [289] and DNA computation [137]. Photonics, with its promise of leveraging light for processing, is believed to offer a solution to the limitations of traditional, electron-based computing. This approach is particularly relevant for energy intensive applications such as AI, where the potential for enhanced performance with lower energy consumption could significantly impact future technological developments.

This chapter provides an overview of optical computing, outlining its fundamental principles, advantages, and current implementations. Optical computing utilises the properties of photons to perform calculations, enabling faster data transmission and processing with reduced energy requirements and heat generation. These benefits are crucial for meeting the growing computational demands in a sustainable manner.

Among the practical applications of this technology, the integration of optical accelerator chiplets into vehicle architectures represents a forward-thinking approach to enhancing computational capabilities in automotive systems. This integration aims at supporting-real time data analysis and decision making, essential for autonomous driving and other advanced vehicle functions.

To summarise, this discussion sheds light on the significance of optical computing in pushing the boundaries of current computing paradigms, offering insights into its potential to revolutionise not just vehicle architectures, but also the broader landscape of computational technologies.

2.1.1 Introduction to using Photonics

Whilst the term of optical or photonic computation is generally not defined to be a specific technology, it is mostly seen as an umbrella term for concepts, that use beneficial properties of light, to gain an advantage over state-of-art electrical processing units in more than one Key Performance Indicator (KPI) that include energy efficiency, throughput and latency.

Some notable characteristics of light of advantage for improving computational performance as a whole are [159]:

Bandwidth

Light, by nature, has a 100,000 times greater usable bandwidth ($\sim$ 500 THz), when compared to electronic circuits (μ5 GHz) [159, P. 5]. This offers optical computation and data transfer to benefit from frequency multiplexing parallelism enabling transmission of several signals on a single hardware medium through differentiation by wavelength channels. A recent experiment demonstrates this principle by achieving a data rate of 1.84 Pbit/s by using 223 wavelength channels on a over a 7.9 km long, 37-core fibre cable [120].

While academia and experiments points out the functionality of photonic computation to be able to work across a wide range of wavelengths [302] (visible, near infrared, infrared spectrum) current experiments and implementations are situated within a range of 1500 nm to 1600 nm [70], [79], [149].

Compute by propagation

The utilisation of unidirectional light propagation through a medium as a mechanism for data processing and computation represents a significant area of interest within the field of optical computing. This approach facilitates operations such as matrix vector multiplication (MVM) and Fourier transformations in linear optics, employing simple optical components like a single lens [159], [181], [247]. The inherent simplicity of one-way propagation offers distinct advantages in hardware design by obviating the need to consider signal back propagation, thereby simplifying the architecture [159].

The concept of leveraging optical components for MVM has gained traction in both experimental and research contexts, leading to the identification of four primary methodologies for hardware implementation. These methodologies, each with its unique set of benefits and challenges, include Multiplane Light Conversion, Mach Zehnder Interferometer, Wavelength Division Multiplexing [302], and crossbar arrays. These options will be elucidated in greater detail in the subsequent section of this document.

A particularly relevant application of these optical computing methods is in accelerating inference tasks for AI. In such contexts, the efficiency of MVM operations is paramount, given their substantial consumption of time and energy [86]. This exploration underscores the potential of optical computing to revolutionise computational paradigms, particularly in AI applications where speed and energy efficiency are critical.

Energy efficiency

Optical computation and data transfers make use of two other advantages, that each contribute to a reduced consumption of electricity.

One being reduced heat generation of optical components and circuits [159, P. 13] [294] reducing the need for active cooling and allows for a weaker air- or water-cooling concept. This is also having the side effect of a reduced room consumption for temperature control, offering a more efficient use of space in areas, where of relevance (*e.g.* instrument panel inside a car or laptops).

The other advantage being the low-loss transmission of optical signals [159, P. 10], that allows for a more efficient use of energy on chip, keeping in mind the issue, that reflections or scattering of light will result in higher losses and an increase of signal noise [159, P. 10].

Optical computation and data transfer technologies make use of two principal advantages that contribute to the reduction in energy consumption. Firstly, the diminished heat generation by optical components and circuits not only mitigates the necessity for active cooling mechanisms but also facilitates the implementation of more subdued air- or water-cooling systems [159], [294]. This effect concurrently diminishes the spatial requirements for temperature regulation infrastructure, thereby enhancing the spatial efficiency in applications where this is of paramount importance, such as within the reduced spaces of vehicle instrument panels. Secondly, the inherent low-loss transmission characteristic of optical signals ensures a more effective utilisation of energy on the chip [159]. It is crucial, however, to recognise that the phenomena of light reflection and scattering can lead to increased signal losses and an augmentation of signal noise [159].

Other beneficial properties of optical computation include:

- Clock speed increase up to several THz [69]
- Programmable and steerable optical beams [159, P. 12]
- Optical data-copying and summation through fan-in and fan-out [159, P. 13]
- Computation at the speed of light [159]

Introduced benefits are of use for different scenarios, that can overall be categorised into clusters of communication or computation. The distinction between them stems from the inherent characteristics and requirements of optical elements in different operational contexts. Data-transfer scenarios emphasise the swift and reliable transmission of information over optical channels, prioritising encoding schemes and modulation techniques that facilitate seamless communication. Communication use cases are already long present with the introduction of Fibre to the Home (FTTH), that were first introduced in 1977 [231] and gained a lot of momentum since then with industrialisation and scaling happening over the world [183] [102]. Optical links are also common within Data Centres [43], where they are used to counter the rising demand of data transfer [44], [121].

Computational use cases involve the integration of optical elements directly into computing processes. With the aim to increase performance and efficiency, research papers and the startup landscape show different methods to achieve a benefit. Efforts range from the idea of fully optical general purpose computation [10] to demand specific accelerators, that use photonic elements to improve a particular use case. Those include vector/matrix multiplication and reservoir computing (within recurrent neural networks, that are yet again based on matrix-vector manipulation) [97], [138], [278], [302] motivated by the increasing demand for performance with increasingly large AI models [89].

The previously highlighted advantages of photonic computation also bring forth challenges in terms of hardware integration and data processing utilisation. Analogue signals by nature are subject to noise, that interfere with the precision of numeric results [63], requiring special care in signal processing and eventually limiting potential use cases for analogue acceleration. The process of encoding digital data onto an optical signal represents a significant challenge as well. The encoding technologies currently utilised for FTTH and broader data communication applications have undergone extensive development and improvement. However, encoding a signal for potential application in analogue computing cannot rely on traditional keying codecs [120]. Depending on their intended application, signal conversion must also be achievable within a very small footprint to facilitate direct integration into a System on Chip (SoC), presenting a formidable engineering challenge, as for processing data through. Signal conversion is evaluated based on two primary performance indicators. Firstly, the precision with which a signal can be read is critical, as it must accurately encode a digital value of x-bits onto the signal and subsequently retrieve it for further digital processing. Secondly, the data rate of the signal conversion is a crucial factor, as it inherently limits the capacity for analogue computation by dictating the volume of data that can be encoded.

Although photonic chips have the potential to achieve clock speeds within the TeraHertz range, incorporating multiplexed signals, practical data rates and clock speeds may be constrained by the efficiency of data encoding and decoding processes. This limitation underscores the importance of advancing encoding technologies to enhance the performance and integration potential of optical computing solutions.

The mentioned challenge of noise is greatly dependent on the actual photonic implementation used for computation making a further analysis of technologies introduced in section 2.1.1 before mandatory. I will briefly introduce the basic concept behind each of the four technologies and address possible limitations of the technology along approximating its noise and effect on precision. The three KPI of interest for implementation and analysis against use cases within the automotive market are the possible numerical precision, bit depth and the method of data loading. The opportunities for integration onto an SoC and setup / load times of data are other things for consideration for a specific service.

2.1.2 Detailed Overview of Concepts

This section gives an overview on concepts that use the idea of photonic computation to enable an acceleration of matrix vector multiplication. Selection of presented technologies is based on the amount of research or academia papers, presentations and usage within the startup landscape and their respective technology readiness level. Noteworthy however, photonic accelerators are still within a very early development phase with limited hardware implementations and a great amount of proprietary information. Information referenced and accumulated within this paper is based on an abstraction level requiring further experiments before implementation and caution with theoretical performance metrics.

Multiplane Light Conversion MPLC

The concept of Multiplane Light Conversion (MPLC) refers to the idea of light propagating through free space with several planes for signal manipulation. Particularly this means modulating light across various planes, each encoded with distinct amplitude and phase information.

Multiplane Light Conversion originates on the idea of Plane Lightwave Conversion (PLC) shown in Figure 1. In the PLC process for optical processors doing matrix-vector multiplication, the incident vector X initially spreads along the x-axis of the system. This is achieved through a cylindrical lens or similar elements, which also replicates X along the y-axis. Next, each element of X $(x_1, x_2, ..., x_n)$is independently modified at the spatial diffraction plane $(w_11, w_12, ...w_nm)$, configured through the transmission matrix W. Lastly, the x-direction beams are coherently combined, culminating in the output vector y $(y_1, y_2, ..., y_n)$ along the y-axis. This output, $y = wx$, represents the product of configuration-matrix w and input-vector x, showcasing the PLC's ability to perform complex computations through light manipulation [159].

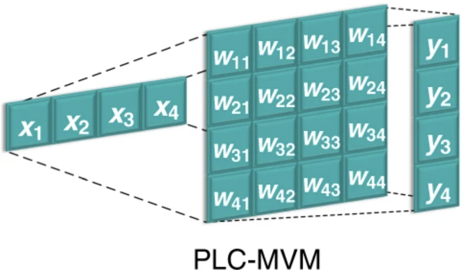


Figure 1: Plane Lightwave Conversion by Zhou et.al. [302]
Sketch of free space PLC implementation using an array of mirrors w_{nm} to manipulate an input vector x_n and steer the beam towards the photodetector y_n.

MPLC builds upon the foundational principles of PLC by introducing the capability to handle multiple optical signal paths concurrently, as depicted in Figure 2. This advancement allows for more complex and efficient processing of data, leveraging the inherent properties of light in an enhanced, multiplexed format.

The light, as it traverses these planes, undergoes transformations, effectively performing large scale matrix operations for matrix sizes up to 490,000 (by using wavelength multiplexing with 700 channels) [302]. Configuring the parameters of used planes offers the use for computation and enables large scale matrix manipulation [42]. MPLC, after its first introduction in the early 1970s [302], is a proven concept in both test setups [238] and experimental tape outs [249], [302]. Whilst large scale matrix multiplication is making MPLC attractive, they require a setup time for reconfiguration of planes for computation making them increasingly slower with changing data per compute.

Data (matrix and vector elements) is encoded through spatial light modulators, that use amplitude, phase, or modulation for analogue representation of a value. The possible bit-precision of each value is therefore dependent on the advancements of modulators, noting that physical noise of the analogue signal (influenced by the medium it travels through) is also limiting the maximum usable precision within an optical signal. An on-chip integration proven to work has shown a precision of 8 bit [302], whilst recent research present an integrated optical unitary processor enabling 12 bits of precision [251].

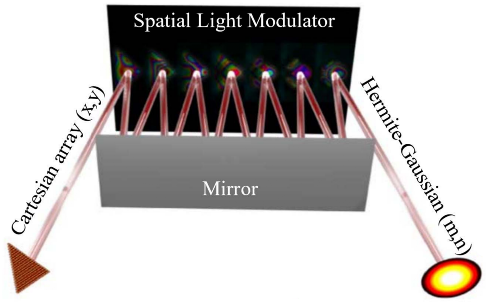


Figure 2: Multiplane Light Conversion by Fontaine et.al. [81]
Example of implementation of MPLC rearranging triangle arrays of Gaussian beams.

Tests for building MVM systems show, that the system noise is dependent on the size of MPLC conversion planes in a linear manner. For a 14x14 matrix the precision of the system was measured to be 98,3% [302].

A research team in 2018 has tested the MPLC method for implementing a purely optical deep neural network and found that, the precision of the all photonic setup delivered worse results, than an all electric conventional implementation. The photonic setup reached a precision of 93.39%, whilst electrical counterparts are expected to reach 99.60% to 99.77% [146].

Mach Zehnder Interferometer MZI

The idea of using a MZI is based on the concept of interference of optical beams, that represent a unitary matrix-vector multiplication and were first introduced in 1994 before gaining a lot of attention over recent years [302]. In greater detail, MZIs are used to perform a rotation of a unitary 2x2 matrix and attentuators are used to scale a specific signal. The concept of singular value decomposition (factorisation method) uses those two concepts of rotation and scaling to enable matrix-vector multiplication, as depicted in Figure 3.

Figure 3a shows how a MZI consists of two waveguides (taking the input signal x_1, x_2) that, after signal manipulation through one phase shifter each (with value θ set from the outside), interfere to result in a completed matrix multiplication [82].

$$U_{MZI}\mathbf{x} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

Whilst a single MZI is used for a linear multiplication of a unitary 2x2 matrix and a 2x1 vector, with the vector mapped to the two optical inputs (x_1, x_2). Building networks requires further attenuators, Those - shown in Figure 3b - use a phase shifter for scaling data values by a factor of $cos(\theta)$.

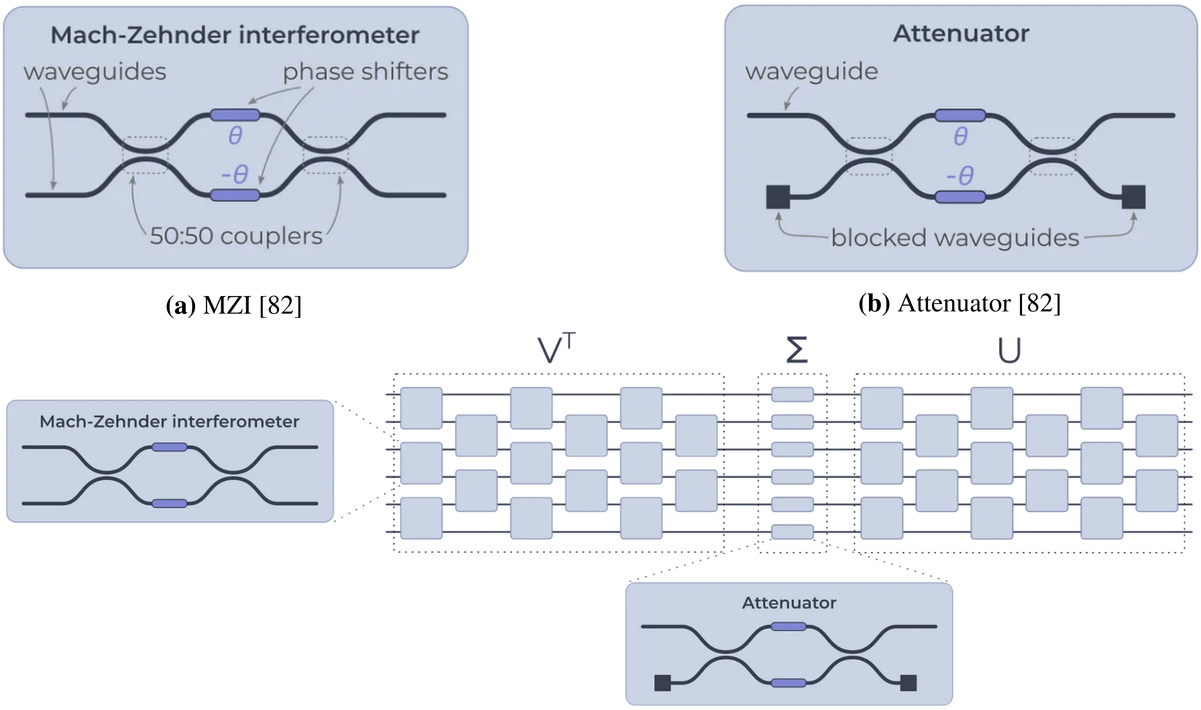
$$D_{\text{attenuator}}\mathbf{X} = x_1 \cos\theta$$

A combination or a network of several interferometers and attenuators allow for multiplication of larger matrices, as seen in Figure 3c.

Whilst maximum sizes are significantly smaller than those used with MPLC [302], research and startup landscapes currently present on-chip implementations of 8 bit precision 256x256 matrix-sizes [64], [236]. Building larger matrix multiplications through a larger network of small multiplication however requires converting "an ordinary matrix into the multiplication of several unitary matrices and diagonal matrices" before processing [149].

Figure 3c represents previously mentioned method of representing a matrix-vector multiplication through two rotations V^T, U and scaling Σ, where the input matrix M is denoted as $M = V^T * \Sigma * U$. The vector to multiply with is an input signal on the left side of this image with the number of input streams being defined through the size of the vector. Shown network (Figure 3c) of MZI and attenuators enables multiplication of a 6 x 1 vector with a 6 x 6 matrix [82].

Digital data is encoded onto an optical signal through phase shift modulators with current technology promising eight [64], [104] or ten bits [221] already integrated on CMOS



(a) MZI [82]

(b) Attenuator [82]

(c) MZI array [82]

Figure 3: hardware schematics of MZI
Overview of hardware implementation of a single MZI (Fig. 3a), a single Attenuator (Fig. 3b) and their implementation for a 6x6 MVM (Fig. 3c).

fabrications. For an optical signal used within MZI, the phase describes the state of a wave oscillation at a particular moment. For instance, a phase of 0 degrees (or 0 radians) might represent the minimum value and 180 degrees the maximum.

Like mentioned with MPLC before, optical signals used within this method are, due to the nature of a physical signal, not free of noise. This once again reduces the precision of the entire systems numerical correctness. Measurements and experiments on noise greatly vary between different publications and are highly dependent of the specific hardware configuration and materials used [223]. A measurement on implemented hardware of company Lightmatter is demonstrating a <0.1% precision loss on an 8-bit 256x256 multiplication based on an array of MZI [279]. Lightelligence, employing a similar approach is is demonstrating MZI-based MVM enabling dimensions of 64x64 [143].

Advancing the here discussed general idea of using a network of MZI for MVM can be done by introducing microring resonator (MRR) to the approach. This enables greater scalability and configuration of the model, potentially decreasing noise within the system [177].

Wavelength / Mode Division Multiplexing

The idea of enabling computation by propagation introduced in Section 2.1.1 is the core concept of Wavelength Division Multiplexing (WDM). Its aim is to utilise different optical wavelengths λ to encode and process information, making use of the inherent parallelism of polarized light [159]. An array of of microrings can be used to enable multiplication of information on optical signals with very little use of physical space and potential for a large number of orthogonal channels [302].

Figure 4 displays a possible design of a WDM implementation. Different input signals $(A_1, A_2, ..., A_n)$, each with their own unique wavelength $(\lambda 1, \lambda 2, ..., \lambda n)$ are created and multiplexed onto a single medium. This combined signal is then split into multiple paths, that will each perform a part of the matrix vector multiplication [138].

An array of MRR is configured to modulate the intensity of a specific wavelength based on each element of the matrix $(B_{11}, B_{12}, ..., B_{nm})$ to multiplicative with. Each row of the MRR array manipulates the row of the array it corresponds to, whilst a single MRR will modulate the signal corresponding to its own column within the matrix. The modulation process effectively multiplies the vector elements (input signals at different wavelengths) by the matrix elements (the modulation imposed by the MRRs). Each modulated signal path represents a summation of these products, akin to the result of a matrix-vector multiplication. Each output $(C_1, C_2, ..., C_n)$ ultimately maps a result of the MVM being the sum of the products of inputs and their matrix weights for that row [138], [159]. Performance of the overall system can, based on the described idea, be increased through introduction of multiple data streams, that distinguish themselves through their distinctive wavelength [138].

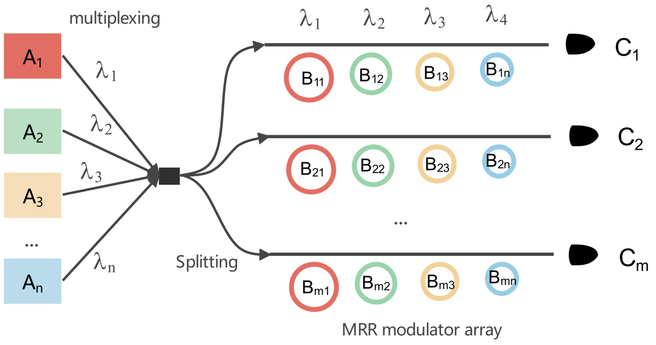


Figure 4: Wavelength Division Multiplexing by Li et.al. [138]
Overview of hardware implementation for MVM using wavelength division and microring resonaters.

The method of using WDM for matrix-vector-multiplication is relatively new in comparison to the two other introduce concepts with first application in 2012. It has however found several uses for various neural networks and accelerators [302]. A major advantage over both MPLC and MZI approaches is, that matrix multiplication requires no previous decomposition of input data for computation.

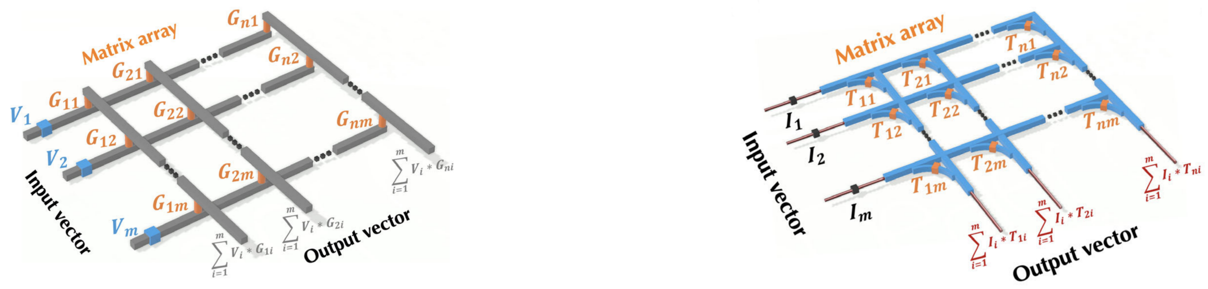
An alternative idea to use wavelength for multiplexing is a concept proposed by Ling, Q. et al., which instead is based on mode division of light. The improvement is a reduction in hardware, as WDM requires multiple lasers for the amount of wavelengths multiplexed, whilst Mode division multiplexing (MDM) is only relying on a single source of light [149].

A major challenge with wavelength or mode division multiplexing is the noise and distortion of optical signals when integrated on chip. Not perfectly tuned mirrors or slight deviance in manufacturing can increase noise as far as making the signal irrelevant for compute. Whilst a simulation of the physical layer yields a model accuracy of 94% in use of the convolutional neural network on the MNIST data set - a decrease of 2% over the standard model [259].

Precision of WDM and MDM approaches are greatly dependent on the detectors at each output. Paper [149] proposes an accelerator of 2-bit precision, but summarises the landscape and academic progress achieving up-to 9bits of precision.

Cross Bar Arrays

A fourth idea to implement MVM through optical processing techniques, is the concept of building a grid of spatial light modulators (SLM). Reflective SLM and transmissive SLM enable encoding of digital information and data manipulation through reflection or transmission of light. The voltage of modulation for electrical crossbars can be denoted as a greyscale mapped onto a bit-level precision. In its simplest form being 1 or 0 for maximum or minimum voltage respectively [78]. The idea of building a grid, that combines vector and matrix information for direct computation is already existent within purely digital concepts with additions in each node of the grid, as seen in Figure 5a.



(a) electrical crossbar array [78]

(b) optical crossbar array [78]

Figure 5: Crossbar MVM implementations [78]
Comparison of purely electric (left) and a photonic alternative (right) of crossbar based MVM showing the mathematical operations realized within hardware.

Figure 5 depicts a sample cross bar array for multiplication of a $[1 - m]$ matrix with a $[1 - m]$ Vector. The "input vector is encoded into a list of the intensity of light waves

(I_1 to I_m) propagating in m optical waveguides" [78, P.2]. The matrix is defined through configuration of the light-transmittance (T_{nm}) within electro-optic components. The photocurrent ($\sum_{i=1}^{m} l_i * T_{1i}$) (electric current through a photosensitive sensor) at each output wave guide is "proportional to the summation and multiplication of input light intensities and tunable transmittance" [78] representing MVM operations. This method can be used to create crossbar-grids to enable larger MVM. Paper [103] presents an simulation of a 256x256 matrix dimension. Paper [178] summarises crossbar-array approaches of dimensions up to a dimension of 100x100.

Hardware level precision (bit level) is based on the advancement of used spatial light modulators's, but can virtually be expanded through a closed-loop approach. Experiments in free-space and in integrated circuits present a highest precision of 8 bit, whilst virtual precision is expected to reach 128 bit without a greater loss in accuracy [31], [78], [103].

Whilst the technology is implemented with startup Neurophos [31] for instance, studies on hardware level accuracy are very limited. A free-space experiment to implement a neural network through cross-bar arrays by [79] showed a decreased model accuracy of 95.3% in comparison to a calculated 96.1%, suggesting a relatively small hardware inaccuracy. A simulation of a crossbar based MVM accelerator demonstrates a setup with a precision of 99% with the claim of measuring no effect on model accuracy or neural network [103].

The four introduced technologies each feature their own challenges and chances. One of the major challenges is the integration into chiplets. Whilst MZIs for instance are already integrated into Peripheral Component Interconnect Express (PCIe) expansion cards, like done by Lightmatter or Lightelligence [67], [142], other methods are yet to be successfully integrated on a piece of fabric. A promising example for integration of free-space-proven approaches relies within the company Optalysys, which has integrated free-space Fourier-Transformation. This is achieved through a vertical chip-extension, as shown in Figure 6. With a one-dimensional design announced in 2023, this integration may show potential for the here introduced concept of MPLC [181].

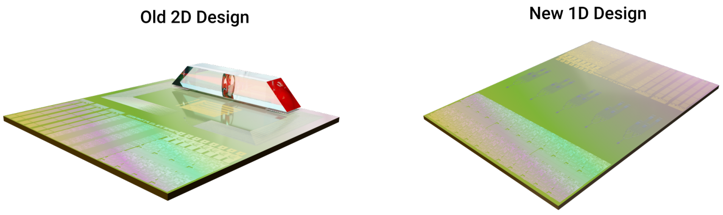


Figure 6: Integration of free-space optical hardware [181]
Example for integration of free-space MPLC within an integrated circuit as proposed by startup Optalysys. First design was two dimensional with a vertical addition to the chip, whilst a new concept promises the possibility to fully integrate MPLC on the PCB

Another element of the question of integration into greater computational architectures, is the hardware size of possible photonic components. With some concepts still being in early research phase with free-space experiment, this question is specific to each approach of photonic data processing further being dependent on the availability of manufacturing capabilities and hardware materials chosen. This topic will further be evaluated in Section 4.3 after an in-depth comparison of photonic approaches.

Noise and distortion of optical signals within waveguides is another major challenge, that impacts the accuracy of the optical processing or MVM in particular. Wavelength Division Multiplexing and Mode division multiplexing are highly dependent on distortion "free" systems, as the usability of the signal depends on the specific wave or mode of the signal. This is especially challenging with both WDM and MDM, as those systems are not re-configurable after manufacturing [138], making integration more complicated due to its "fragility".

2.1.3 Summary on optical Computation

Previous analysis of four approaches to optical computation and held discussions with startups show that optical computation can be beneficial for specific use cases. As presented by research, those include data transfer (and related switching methodology) or specific mathematical operations such as MVM. Optical accelerators may impose a great benefit to use cases, in which:

- Highly repetitive workload with similar / specific algorithms (with incorporation of MVM operations). The knowledge of algorithms used enables the design of customised hardware to gain a significant advantage over digital specialised hardware or digital general purpose processors.
- High throughput by multiplexing of the analogue signal to enable multiple computations on the same hardware in the same clock cycle.
- Energy efficiency through beneficial properties of light.
- Resilience to noise / reduced numerical precision.

Two drawbacks must however be considered in using optical components for computation. Analogue signals are not immune to issues in numerical precision due to the medium they are transferred through and the manipulation of such with modulators within processing chains. This results in a negative effect on the numerical correctness of the result of a computation. In simple terms, $4+4$ must not return the value 8, but a result distributed around the correct value. Another drawback is the representation of digital values on an analogue signal. Hardware development is limiting the maximum achievable precision and might further increase the impact of analogue noise on numerical precision.

Table 1 summarises of all four photonic approaches previously presented and depicts the major KPI with the currently enabled bit-precision, matrix dimensions and accuracy of the system.

Table 1: Different optical technologies and their performance

Method	Precision (bit)	Matrix Size	Accuracy	Data-Loading
MPLC	12	490000	>99	Single-load
MZI	8	256	>99	Single-load
WDM	<9	25	98	Single-load
CrossBar	8 (128)[1]	>2	>99	Single-load

[1] virtual precision

A topic left out of previous considerations is the respective energy consumption of each technology. Whilst a significant improvement due to the inherent usage of photonics (instead of electronics) is explained in numerous papers [64], [82], [159], [302], the effective gain is dependent on the specific approach implemented and use case services. Zheng, S.N., Zou, J., Cai, H. et al. compare the energy efficiency of MRR (35 mW each) and MZI (1.8 W each), showing how a MRR achieves $\sim 5x$ greater efficiency [300] over an MZI approach (accounting for the direct MVM part of the system only). An advanced and optimised combination of MZI and MRR are reportedly able to realise a non-linear activation function (combination of an MZI, MRR and two heaters) with a consumption of just 1.3 mW [116]. Additionally, paper [250] depicts an MPLC approach with 8.4 mW for each phase shifter. These numbers give an insight into the respective energy consumption of specific parts of the value-chain within a photonic accelerator. With little knowledge on the amount of needed MZI or MRR components for realisation, those will have to be put into greater context and comparison to purely electric hardware accelerators in Section 4.4.

To summarise: The current landscape described in research, academia and the Startup landscape leads to the overall assumption, that optical computation is expected to enable efficient computation with high throughput for use cases where an algorithm, that greatly uses matrix-vector multiplication is used. A benefit can however only be realised when the service accelerated is resilient to noise and is not dependent on 100% numerical precision. The category of AI-inference use cases for instance might be suitable, as a minimal reduction of accuracy may not make a huge difference for its application [86]. Use Cases and the impact of hardware noise will further be discussed within Chapter 4.1.

2.1.4 Ongoing Industrialisation and Integration

The concept of using beneficial properties of light for computation isn't entirely new and been around for several years already [78]. Improved possibilities for fabrication and the increasing demand for AI computation use cases and energy efficient processing however have greatly increase the interest in such [159]. Ongoing research has brought up several startups over the last few years, that this Chapter is giving an overview on with the idea of giving an overview of current products on the market. Companies were identified through search-engines, conferences, papers and journals.

United-States based Lightmatter and Lightelligence both follow a similiar approach after their founders co-published a paper in 2017 proposing an optical neural network to accelerate AI use cases and reduce heat generation [226]. Both startups build an array of MZI to enable MVM on an accelerating chip and currently offer a ready-to-integrate product, that integrates through the PCIe slot of a motherboard or comes as a 4U form factor server-module. Whilst compute-able matrix-sizes differ, both advertise the benefits of energy-efficiency, throughput and no heat-generation [82], [142], [144]. These claims directly link to earlier introduced benefits of optical computation in Section 2.1.1.

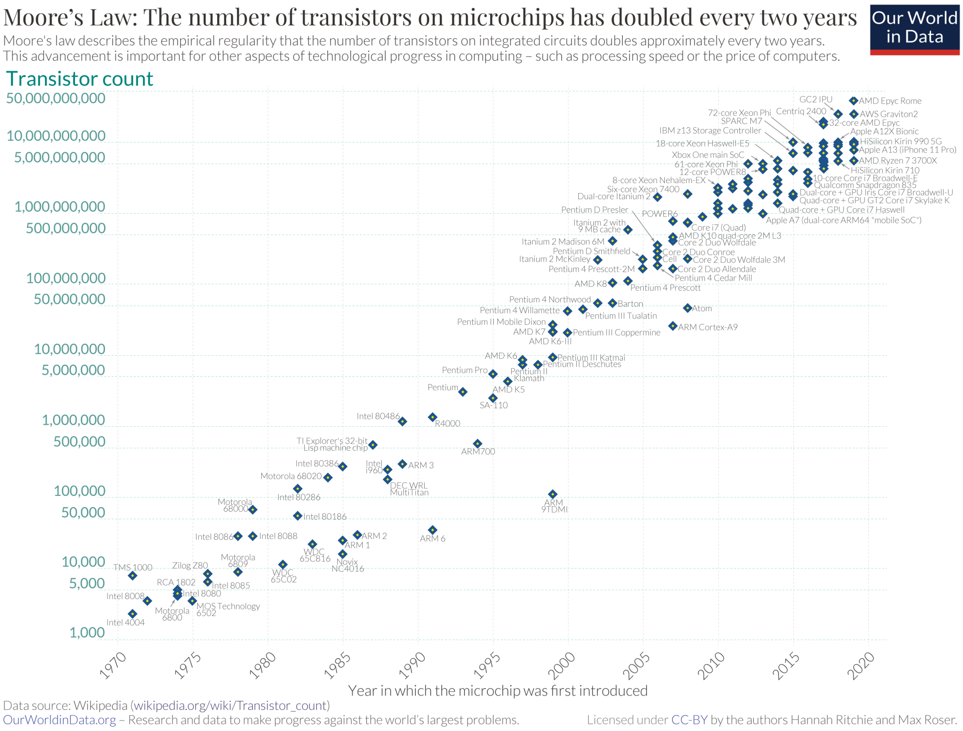
Several other startups are delving into the usage of photonics, that include Akhetonics [10], [91], Luminous [35], Cerebras [38], Celestial AI [9], Salience Labs [26], Cognifiber [51], 3E8 [1]. Public knowledge on product offerings and roadmaps of these companies is limited, but few trends are visible, that include a great focus on data-transfer / interconnects, the acceleration of AI use-cases and the goal of a new generation of chip-development. Akhetonics stands out with a wider goal of enabling general (XPU) acceleration units.

2.2 Development of State of the Art Processors

Ever since the first processor was introduced in the 1970s computational power has been a major enabler for innovation across all industries [77]. The development of processors has become a crucial factor in the advancement of both technology and society, whilst the need for more "power" is a constant ever since. The goal remains to enable faster processing, greater throughput at a possibly small energy consumption. Several technological advancements over the last years have led to the current state, in which everyday life is surrounded by handheld computing supporting humans at all times.

This Section will give a short, high-level overview of advancements and trends in computing over the past years to highlight different technological enhancements leading to a greater performance overall. Those enhancements include, but are not limited to, the ongoing miniaturisation of hardware, increased packaging and modularization and advanced architectures [167], [224]. An understanding of such will then enable discussing current developments, ideas and limitations for the upcoming years of processor development.

One of the key-changes on processors throughout the years is the sizing of hardware components. The Intel Pentium III Coppermine processor introduced in 1999 was manufactured on a newly introduced 180 nm node [112] for instance. A node refers to the technology used for fabrication of integrated circuits indicating the smallest half-pitch of contactable features. The successive Intel Pentium IV has featured a smaller 90 nm node just 5 years later being an example for the pace of miniaturisation. Current times now feature various processors manufactured on a 3 nm node including the current generation of Apple Mac and Phone processors [154]. Production of 2 nm nodes are currently on the way with a market entry expected in the later 2020s [58]. Smaller nodes enable more transistors on the same area / die size, thus offering space to more ALU or caches for increasing theoretical computational throughput per clock-cycle. The increase of transistors per chip is approximated through Moore's Law being an empirical observation, that the number of transistors in an integrated circuit doubles approximately every two years. It was first described by Gordon E. Moore, the co-founder of Intel, in 1965. Moore's Law is not a law of nature, but rather an observation over a long-term trend in how technology is changing. The observation seen in Figure 7 has held true for more than half a century, driving linear growth in the number of transistors on integrated circuits and leading to significant advancements in computing power and capabilities [167].


Moore's Law: The number of transistors on microchips has doubled every two years

Moore's law describes the empirical regularity that the number of transistors on integrated circuits doubles approximately every two years. This advancement is important for other aspects of technological progress in computing – such as processing speed or the price of computers.

Data source: Wikipedia (wikipedia.org/wiki/Transistor_count)
OurWorldinData.org – Research and data to make progress against the world's largest problems.
Licensed under CC-BY by the authors Hannah Ritchie and Max Roser.

Figure 7: Transistor count over time known as Moore's Law [211]
Overview of increasing amount of transistors on microchips over the year with a clear trend visible as foreseen by Moore.

The prediction of Gordon Moore, that proved itself to be true for years, has become an increasing topic of discussion and controversy over the past years. It is greatly discussed, when and if Moore's Law will end [167], [224], [282]. Whilst NVIDIA Chief Executive Officer (CEO) Jensen Huang for instance has repeatedly mentioned Moore's law to be dead arguing that "the method of using brute force transistors and the advances of Moore's law has largely ran its course" [50]. Intel CEO Pat Gelsinger is calling the law "alive and well" in 2022 before talking about a decrease in the doubling rate time to 3 years (from previous 2 years) in a recent speech [11], [50]. The question of an end of Moore's Law is also present in research and academia with past years bringing up several papers discussing, if the current produce able node would be the commercially smallest achievable [282]. Ironically, those papers have quite often already proved themselves wrong with smaller nodes (3 nm) already being in mass-market production [154] and with Intel recently announcing building a new manufacturing site in Germany for fabrication of 1.5 nm chips [229].

A common theme and reasoning for Moore's Law ending amongst papers and conferences on the possible end of increasing miniaturisation of transistors, is an inevitable physical limit [208], [282]. Several suggestions of a physical limitation have already been proven wrong in either experimental or commercialised state. Current research suggests, that a physical minimum gate length might be situated at 0.34 nm; the size of a single carbon atom [208] - currently considered to be a minimum viable size, whilst not proven for commercial use in millions of units just yet [71]. While a physical limit to the pure size of chips may be coming up, ongoing engineering efforts are expected to enable continuation of Moore's Law for a few more years, with sub 2 nm nodes being in active research with hopes for commercialisation in the 2030's [267]. Decreasing transistor sizes however implicate increasing challenges for manufacturing and energy consumption making the discussion not just a technical but a business decision as well, as size is not the only possible option for improving performance of processors [267], [282].

Two other major ideas in increasing computational performance are improved processing architectures [135], [224] and 3-dimensional packaging of chips [224]. The term of a processor architecture is a broad one, that can characterised by the orchestration of both hardware and organisation of a processor, as described by Hennessy, and Patterson [100]. The organisation thereby includes a high-level design of storage, processing and data transfer, with each of the three being influenced by the Instruction Set Architecture (ISA) chosen. The ISA describes specific operations executed on within the processor. They can be categorised into two types of designs with an either more complex set of instructions Complex Instruction Set Computing (CISC)) or a more compact set (Reduced Instruction Set Computing (RISC). Whilst picking one of two has an impact on the design of a processor, there are specific implementations of each category introducing their own flavours like ARM and RISC-V for the category of RISC [25].

An option to further accelerate computational performance is the introduction of specialised hardware accelerator chips. Accelerator chiplets are application-targeted hardware implementations, that instead of being able to universally compute, are meant to efficiently solve a specific type of algorithm - a concept known as an application-specific integrated circuit (ASIC). This enables higher performance for the same power-consumption of a chip. [83], [90]. This is further motivated by the computational demand of battery-run devices and the increasing emergence of compute-heavy applications like AI [83]. Using accelerators is now considered state of art and has found its way into a variety of use cases and hardware-environments. Google for instance has developed Argos as an accelerator for video encoding, that is core to the YouTube service [46]. Built to be integrated into existing server-racks, Argos is embedded on a PCIe expansion card and features a total of ten custom video encoder cores specifically designed to encode input-videos in H264 encoding enabling higher processing power and lower consumption of electricity [186].

Similarly to the idea of Google to design an ASIC specifically for YouTube data-centres, other chip-manufacturers have followed similar ideas for non-data-centre applications. Examples are the introduction of an NPU into Apple devices [122] or Snapdragon [237] processors.

Mentioned processors make use of another trend, that began in the 1990s and has since then been greatly accelerated through increasing numbers of mobile and battery powered devices [108], [209]. A System on Chip (SoC) is considered as a combination of hardware blocks required for processing data on a single integrated circuit. It consists of at least some sort of microprocessor, but usually contains an on-chip memory and a communication link [219]. Already mentioned Apple M chip-series for instance is a great example for a modern SoC made up of unified memory and several computational units specialised for the use within a portable laptop. This can be a GPU, NPU, specific I/O or other use-case specific accelerators, like previously mentioned video-encoding unit [15], [122]. The increasing affordability of SoCs, their promise of enabling low-power, high-efficiency and miniaturisation of hardware components have lead to them being a state-of-art for domains such as mobile and automotive [16], [201]. The technical possibility and advancement of multi-die integration and 3-dimensional stacking is further increasing opportunities to SoCs [134].

The integration of diverse hardware components onto a single System on Chip and the strategic inclusion of hardware accelerators embodies the essence of heterogeneous computing. This paradigm advocates for the synergistic utilisation of specialised processing unit (PU) to optimise task execution, leveraging the distinct capabilities of each unit tailored to specific algorithms at a given moment. The principle underpinning this approach is the exploitation of the unique strengths inherent in a individual processing unit, which are characterised through their architectural design and operational specifications tailored to a specific service to be used for. These characteristics are significant in determining the suitability of each PU for various computational tasks. When effectively orchestrated, this methodology can substantially enhance the overall utilisation of hardware resources [163].

Mittal and Vetter underscore the critical role of adept scheduling and workload partitioning, predicated on the comparative performance of each PU, in maximising processing efficiency and energy conservation concurrently [163]. The primary challenge in employing multiple PUs for a singular task or algorithm lies in the management of data transfer and synchronisation among the involved hardware components. Inefficiencies in scheduling and memory provisioning can render a hardware accelerator redundant in terms of performance (or at least significantly impact their effectiveness), as demonstrated in their work.

The concept of deploying SoC with heterogeneous processing units, each complementing each other's capabilities, is not a novel notion in both consumer electronics and embedded systems. The Apple M1 Silicon and its successors [15] illustrate the practical application of specialised units within an SoC for distinct domains, such as AI inference and image processing. Similarly, the Snapdragon 888 SoC, integrated into devices like the Samsung S21 series [200], amalgamates a CPU, GPU, digital signal processor for AI applications, and an image signal processor, showcasing the effective use of hardware accelerators for specific use cases [199]. The Tesla FSD chip or the integration on an NPU within a Mercedes concept are known examples of the concept within the automotive sector [254], [277]. All previous examples are based on the ARM architecture.

Intel's introduction of the Lakefield processor at Consumer Electronics Show (CES) 2019 was considered a significant step into heterogeneous processing for the x86 architecture, incorporating specialised hardware components such as low-power cores [59]. Despite its considered lack in performance (in comparison to similar x86 state-of-art processors) and premature discontinuation two years post-launch, the venture into x86-based heterogeneous computing has garnered significant interest [148]. This interest has been further propelled by chip manufacturer AMD's recent unveiling of a new SoC component (Ryzen AI) targeted at consumer and automotive sectors. This SoC embodies the principle of hardware acceleration for AI and enhanced digital customer experiences, marking a pivotal advancement in the domain of heterogeneous computing [19].

While heterogeneous computation was limited to using technology based on the same die-size and therefore limiting the possibilities for integration of hardware -accelerators, ongoing research show advancements for multi die size integration. This enables more flexible and efficient design approaches being the foundation for the combination of different types of processors within a single package, like photonics and conventional electronic processing [41], [134].

To summarise reviewed developments and trends in development of processors or SoCs in a broader scope; Increasing effort, mandatory to continue improving computational performance and overcoming dawning physical limitations in miniaturisation of processing hardware builds growing interest in alternative methods to improve both performance and efficiency of next generation processing units. Advancement in manufacturing thus enabled enhanced packaging methods will complement superior processor architectures according to Gordon E. Moore in 2006 [167]. Both assumptions are now considered state of art and have greatly been improved by the rise of mobile applications and battery-powered devices. System on chips are now common for many types of devices and feature hardware accelerators to increase performance and efficiency for specific domain computing.

3 Application within Automotive

This Section of the document will give a high-level overview of computationally demanding services and features within a vehicle. This includes both passenger driven and safety driven products. Whilst the list has no intention to be complete, it is meant to point out services of great computational effort that could potentially be improved through optical hardware acceleration. Standard workloads like UI loading, vehicle-switches (like window-openers) and such are not included due to their low computational demand and their implementation status. The Section will then enable the identification of certain services within the vehicle that may gain most advantage from acceleration.

With the rising demand for digital services and features within and around the car, efficient computation and the distribution of workload is of rising interest for several years already. A service within the vehicle can be defined as a piece of software, that offers some sort of functionality of value for the customer throughout various domains. A certain requirement of a specific service is known as a Service Level Requirement (SLR). The 5G Automotive Association (5GAA) has proposed a classification of use cases within 7 groups for the purpose of defining SLRs for Vehicle-to-Anything Communication (V2X) in efforts of discussing the communication of vehicles [2] [3] [4]:

- Safety
- Vehicle Operations Management
- Convenience
- Autonomous Driving
- Platooning
- Traffic Efficiency and Environmental Friendliness
- Society and Community

Whilst this categorisation highly generalises infotainment (information and entertainment) use cases into convenience and adds complexity to services around the software-controlled vehicle movement and fleet orchestration, it is a reference to analyse possible applications of optical computation by their SLR. To simplify analysis of use cases, I will refer to the following use cases categories and then begin with a collection of sample services through interviews with colleagues and an active scouting of existing vehicles:

- Infotainment
- Vehicle Safety
- Vehicle Control

Infotainment

The cluster of infotainment use cases summaries services, that serve both entertainment and information of vehicle passengers and create value or convenience to customers. It encompasses multimedia, navigation, custom applications, and other features designed to keep drivers and passengers entertained and informed whilst on the road. Use cases of automotive infotainment include in-vehicle entertainment (music, video, gaming), navigation services (traffic information, directions), vehicle information, smartphone integration and more. They often rely on connectivity for communication with the back-end to enable streaming of data and handling of customer-requests, dependent on localisation of information of interest.

Vehicle safety

Vehicle safety clusters use cases that ensure the well being and health of passengers without software being in charge of vehicle control. These measures may go beyond conventional safety features, incorporating solutions that contribute to accident prevention and mitigation. They include Collision Avoidance, cross-traffic alerts, emergency response, traction and stability control.

Vehicle control

Vehicle Control use cases are those, that take main responsibility over vehicle movement for at least a short amount of time (*i.e.* $\geq$ 10s for level 3 ADAS). Those use cases range from interventions in specific situations (like steering the vehicle back into its designated lane) to the use of ADAS. ADAS must however further be distinguished in the five different levels of software- or OEM-responsibility for the actions of the vehicle. At Level 0, there's no automation; the vehicle is entirely controlled by a human. Level 1, or Driver Assistance, introduces systems like adaptive cruise control, where the car handles some tasks but still requires active human supervision and takes no responsibility over vehicle actions enforcing constant passenger readiness.

Partial automation in level 2, sees vehicles managing both steering and acceleration under certain conditions, yet the driver must remain able to intervene within a second. Level 3, conditional automation, advances partial automation, allowing the car to take full control in specific scenarios (*e.g.* highway sections), with the driver ready to take over within a maximum of 10 seconds [75]. This means, that at any given time, the vehicle must be able to remain in control for the next 10 seconds moving responsibility for those seconds from driver or passenger to the OEM. Level 4, high automation marks the stage in which a vehicle is capable of performing all driving tasks in certain environments without human intervention or supervision. Finally, full automation is achieved with level 5, where the vehicle is completely autonomous under all conditions, rendering the need for a human driver and steering wheel within the vehicle obsolete [27]. The specific level of automation of a vehicle is having a direct impact on the requirement in terms of reliability onto the entire processing chain of the driving assistance systems, ranging from hardware to software.

Using the service categories one can identify specific use cases to further analyse. Each of these can be described by different SLR. Three requirements are particularly of interest for determining the possible use of optical computation for each use-case.

- Service Availability

- Service Latency
- Data processed / Throughput

The service availability is a measure, that describes the percentage (%) of time on a vehicle journey, in which a service must be available. Percentages can be clustered into three categories respectively. Services, that are a convenience to the customer, but not relevant for safety should be available 95% of the time. An example for those would be the offering of video streaming - whilst irrelevant to vehicle movement, it is considered as a minor issue or disturbance when not available. Services, that impose a major inconvenience to the customer, when not available, but not relevant for vehicle safety should be available at least 99% of the time. A great example for this scenario is the offering of navigation, that greatly interferes with the customer satisfaction. Software, relevant for the actual vehicle safety must be functional for the entire trip. The categorisation into 95% or 99% is derived from the 5GAA whitepaper [3] and conference presentations using this schema for classification. As availability of 100% cannot be guaranteed [180], the requirement for service crucial for safe operation of the vehicle must be available for at least 99,99999999% of the time, as defined through ISO 26262 in Automotive Safety Integrity Level (ASIL) class D [195]. This cluster includes all services that take control of the vehicle for a specific amount of time. This can either be short-term like Dynamic Stability Control (DSC) or over greater amount of time within level 3, level 4 or level 5 autonomous driving. The risk classification ASIL is a great reference to specify service reliability, as the ISO 26262 categories vehicle features into safety-criticality [232].

Latency refers to the delay in milliseconds (ms) for a service to respond to a certain request or more. Generally spoken it describes the time passed between the "occurrence of the event in scenario application zone [and] the beginning of the resulting action" [2, P. 7]. The exact definition is dependent on the use case analysed. For infotainment purpose this can be seen as the between user-input (actuation) and information displayed on a medium within the vehicle. For ADAS this time can be identified as the processing time of data into information used for further action. Latency requirements can result from various factors. In infotainment purposes latency is critical to crate a seamless user experience that doesn't feel "laggy" or unresponsive. Vehicle movement services have a latency requirement, that is dependent on the situation of the vehicle. Higher velocity requires smaller latency's to allow a safe operation and response to spontaneous events on the road ahead.

The requirement of data processed is outlining the data throughput of a specific service in Megabit per second (Mb/s).

3.0.1 Infotainment Use Cases

Services in this cluster are mostly seen as non critical to customer satisfaction, but being conceived an inconvenience. If they are unavailable for a small amount of time within a user journey, they do not interfere with the users intention to drive or be driven from source to destination. Preparation of user information and the display of a graphical user interface is not part of the list, as it is a very general use case without specific algorithms or applications. A previously undefined criteria the positioning of computation (onboard vs offboard) is the origin of content, that can either be produced on the fly, dependent on user input, or be pulled from external sources or cloud providers. Video streaming for instance is a direct integration of vendors external clouds, that serve a stream of data (like Amazon Prime, Netflix, YouTube or Disney). Rendering of videos on the other hand might depend on live-data from the vehicle, when taking ideas like Extended Reality (XR) into account.

Table 2: Infotainment services and their SLR

Service Name	Availability (%)	Latency (ms)	Data Throughput (Mb/s)
Video Streaming	95	100 [1]	250 [2] [3]
Music Streaming	95	100 [1]	1,5 [3]
In-Car Gaming	95	20 [3]	20 [162]
Navigation	99	1000	0.009 [74] [4]
Video Rendering	95	100	>2000 [155]
Personal Assistants	99	<1000	0.673 [5]

[1] latency of network stream assuming limited buffering

[2] estimation of single 8k video stream

[3] free lossless Audio Codec (FLAC) stream assumed

[4] Assuming off-board route calculation. The datarate is assumed through GMaps network requirements [74]. In-vehicle route calculation would enlarge data throughput and processing time significantly and require the vehicle to store or pre-load the map for the entirety of the route ahead.

[5] Alexa, as a reference, is utilising a cloud based approach to language processing and response generation [93]. Datarate is approximated with a high-quality microphone that records audio at a sample rate of 44.1 kHz and a sample depth of 16 bits. Whist this quality is likely to exceed the required quality of a vehicle.

3.0.2 Vehicle Safety Use Cases

Safety equipment and services of a vehicle are a vital part to ensure the well-being of a passenger and are required for authorisation of sales. Services named below are therefore categorised as ASIL-D through ISO 26262, meaning that they require a failure rate of less than $1E-10$ /hr [206]. Accounting for an availability of 99,99999999% in percent.

Table 3: Vehicle Safety services and their SLR

Service Name	Availability (%)	Latency (ms)	Data Throughput (Mb/s)
DSC	99,99999999	100	<1 [1]
ABS	99,99999999	100	<1 [1]
Airbag Control	99,99999999	<25	<1 [2]

[1] Antilock Braking System (ABS) and Dynamic Stability Control (DSC) use a variety of sensors to compute action of engine control and braking system. Those sensors include wheel-speed, yaw-rate, accelerometer, steering angle and brake pressure that provide a data stream each. Assuming a representation per value in the size of a float16 and a rate of 50Hz per sensor would result in a data rate of 800bit/s per sensor accumulating for 9.6 kBit/s for a setup of twelve sensors within the vehicle. Whilst the exact number of sensors used for both systems are unknown and greatly specific across vehicles and OEMs, it is expected, that the total data rate would not outgrow 1 Mbit/s.

[2] The perfect timing of airbag inflation requires great precision and accuracy for inhabitant protection, thus creating the need for sensors with higher sampling rate compared to previously mentioned sensors for ABS and DSC [107]. Using the assumption of a float16 for value representation and figuring an increased sample rate of 400 Hz [5] yield a data rate of 6.4 kBit/s. With an estimation of 4 sensors for vehicle sides and 3 sensors each for the vehicle front and back, that would accumulate to 64 kbit/s for a simple sensor based collision detection. It is notable, that the system calculated is a reactive system with the meaning of extremely fast response to a crash. Enhanced systems for passenger protection would use a greater set of sensors to predict crashes and take pro-active action, like increasing the force on seat-belts. This idea is, for simplicity reasons, seen as part of environment perception in the next chapter.

3.0.3 Vehicle control use cases

Unknown latency requirements are approximated by calculation of the covered distance at a velocity of 130 km/h. 20 ms at that speed account for 0.722 meters covered - 100 Milliseconds would cover 3.6 meters. The maximum latency for ADAS use cases within the automotive industry is commonly discussed at 100ms to enable sufficient time and distance to react to unforeseen events [152, P. 17]. The required availability and reliability of a specific service is dependent on the level of autonomy the vehicle will be in at a given moment of time. The level of autonomy dictates the time, a user might need to take over vehicle control. The smaller the time of manual intervention, the more situations must a car master and the less frequently should a system malfunction or be unavailable, in general. With the drivers attention moving away from actively monitoring the vehicle and the need to possibly take over control from level 3 onward, the system itself must be fully reliable in the situations it is expected to master (*e.g.* controlled highway environments). This, similar to previously mentioned safety use cases, corresponds to the ASIL category of failure rates and responsibility requiring ASIL-D failure rates.

Table 4: Vehicle control services and their SLR

Service Name	Availability (%)	Latency (ms)	Data Throughput (Mb/s)
Environment perception	99,99999999 [1]	100	>18500 [3]
Vehicle Path projection	99,99999999 [1]	100	>6000 [4]
Lane departure warning [2]	99	100	4600 [4]
Blind spot assist warning [2]	99	100	4600 [4]
Traffic Sign Recognition	99,99999999 [1]	200	>4600 [4]
Driver Health Monitoring	99	200	2000
Lane Keeping / Assist	99,99999999 [1]	100	4600 [4]
Adaptive Distance Control	99,99999999 [1]	100	500

[1] for ADAS level 3 and above due to responsibility of the system moving to the OEM of the vehicle

[2] as an optional customer service, that is not part of environment perception needed for autonomy of the vehicle

[3] This metric is based the amount of sensors on a car needed for a total perception of the vehicle combining Light detection and ranging (LIDAR), RADAR and camera based vision (with possibilities to further extend sensing setups through IR-cameras or V2X). An exact number of sensors required for full coverage of a vehicle's surroundings remains undefined and differs greatly among various manufacturers. Tesla for instance, being perceived a market leader for autonomous driving, is currently deploying its Full Self Driving (FSD) hardware with a total of eight cameras featuring a resolution of 2896x1976 pixels and a maximum frame rate of 40 Frames per Second (FPS) each (with a 10 bit color depth) [57] [56] [55]. This results in a total data generation of 2.28 Gbit/s (rounded) to be processed per camera and a total of 18.3 Gbit/s for processing data from all cameras, needed to enable overall environment perception. Other manufacturers, like BMW or Mercedes are including LIDAR and RADAR sensors within its sensing setup to enable improved object detection in challenging environments. The metrics of importance for such sensors are considered to be the amount of points per second (PPS) and the information depth for each point usually consisting of at least relative position and reflectivity. A sample radar introduced in 2023 offers the feed of 5,242,880 PPS with a data rate of 254.3 Mbit/s [110]. The recent Life Cycle Impulse (LCI) of the BMW 7-series in 2023 has introduced a LIDAR with a data feed of 10 million PPS, potentially resulting in 550 Mbit/s per LIDAR [202]. BMW is further fitting the mentioned vehicle with ultrasonic sensors, Radar and an unspecified amount of cameras for level 3 ADAS readiness. With the minimum amount of sensors unknown for level 4 automated vehicle control, I assume, that at least 2 LIDARs

(forward and backward facing) and a combination of 8 cameras are needed to fully recognise the environment around the vehicle. This assumption is based on conference presentations and references as the NVIDIA setup proposals [22], [175]. Assumed sensors account for a total data rate of 18.5 Gbit/s.

NVIDIA being a selected tier 1 supplier by several OEM (including Mercedes, Volvo / Polestar, Hyundai, XPENG and Jaguar Land Rover) for ADAS systems [176] is proposing a larger setup of sensors containing: 15 cameras (8x 8.3 MPx external, 4x 3.0 MPx external, 3x 5 MPx internal), 9 radars (mix of long and short range), 2 LIDARs for imaging [175]. Assuming previously used 40 FPS with the native raw bit-depth of each camera, the total data generation of all 15 cameras would equal a total throughput 35.33 Gbit/s, marking a significant increase over the assumption above.

[4] Assumption, that two cameras and GPS / map data are used for service realisation. Map data with detailed information of traffic lanes and speed limits can be used for proactive vehicle steering and preparation. Cameras or LIDAR function as real-word reference and fall-back solution if map-data is unavailable. Using the previously mentioned camera with a resolution of 2896x1976 pixels at 40 FPS a data throughput of a minimum of (2.29 Gbit/s per camera) is assumable.

3.1 Analysis of Use Cases

The collection of services and their SLR beckons the opportunity to identify potential candidates, that might benefit from an integration of optical accelerators within the vehicle rather than a potential offload to back-end services. Comparing previously analysed benefits of optical components (in Chapter 2.1.3) with services, enables the direct detection of services, that might benefit from optical computation.

Connectivity plays a major role in the offload of services to cloud architecture, as it acts as a key-requirement for moving hardware computational resources for a service of specific availability requirements.

When the vehicle's connectivity status, measured in terms of availability percentage, exceeds that of service availability, and the expected latency surpasses the actual latency experienced, it becomes feasible to entirely host services offboard (*i.e.* cloud). Surveys from the European Space Agency (ESA) show, that 92.9% of type A and B roads in northern Germany (federal motorways and federal highways) are served with broadband connectivity (4G or 5G) [14]. The test setup for this measurement however, was a smartphone placed within the vehicle and therefore represents worse results, than high-gain antenna setups of the vehicle itself, placed outside of the interior and experiences less distortion (due to infrared-shielding and metal components). Another measurement on rural roads encompassed within the "Tour de France" in 2023 has shown, that 62,3% of the journey was experiencing "poor or lacking" (signal strength < -104 dbBm) internet service [241]. This test was yet again conducted with a smartphone within a vehicle, that is also fitted with an Infra-Red (IR) coating further reducing the signal strength in comparison to vehicle antennas. It is expected, that with current actions taken towards a better connected vehicle through advancement in terrestrial connectivity and the ongoing development of satellite-based connectivity [14], [28], [29], [240] an OEM reaches its customers on a global scale with a latency of <200 ms and a mediocre data rate of >10 Mb/s with an availability >95%.

The previous Section enables insight into the three selected use case categories for infotainment, safety and vehicle control. Table 2 lists five sample services with rendering of graphical videos standing out from the rest with significantly larger data throughput, than simpler information display (like navigation information) or streamed media Assuming a stable and reliable internet connection of the vehicle enables offloading of navigation and personal assistants, as data rates are low and latency requirements non-critical. Rendering video content marks an exception with increasing amount of in-car displays and a great amount of pixels to be rendered, making a possible acceleration interesting.

Services within the safety category summarised in Table 3 are categorised through their low data rates and latency's needed for safe vehicle operations.

Lastly, vehicle control services seen within Table 4, all feature high data throughput's (considering raw sensor feeds) and low latency requirements due to safety aspects. Environment perception is particularly standing out as an enabler for increasing levels of vehicle autonomy and its great amount of sensor (*i.e.* camera, LIDAR, RADAR, ...) data to process.

The analysis above shows that optical computation, from a SLR viewpoint, might be helpful for two use case clusters. One being, due to its data throughput and latency requirements, the perception of a vehicle environment used in ADAS systems and the second one being rendering of 3D-environments for eventually coming application of XR or already existent 2-content within the vehicle.

4 Analysis of Opportunities for Integration of Optical Components into Processors

4.1 Further analysis of identified Use Cases

This Chapter delves into a deeper exploration of services previously pinpointed for a potential benefit from optical computation in Section 3.1. It provides a comprehensive analysis of selected fields of interest, specifically environment perception and video rendering. Each of those areas are examined in greater detail covering their requirements, potentials, and projected future demand. This will offer a thorough overview of the algorithms and mathematical operations used, as well as the method of data processing. Doing so will enhance the understanding of a possible benefit from photonic computation and allow initial insight into design objectives for a potential optical accelerator.

4.1.1 Environment Perception

Use cases within the cluster of ADAS are standing out from other use cases within the category of vehicle control because of their high data throughput and safety-requirements. Environment perception, being an enabler for autonomous vehicles is particularly interesting because of the great amount of sensor-data processed. Other features like path-projection, blind-spot warnings or traffic sign recognition may be of similar importance for security but use a subset of the sensors used for environment perception. Vehicle-to-vehicle communication is another potential "sensor" for building a digital model of the vehicle surroundings, as other infrastructure users could stream their location, metadata or own view of the world - this input will not be part of the following overview for simplicity reasons.

The aim of a complete environment perception setup is to build a contextual understanding of the vehicles surroundings. This includes both detection, segmentation and prediction of objects, as well as the categorisation by semantic meaning of a tracked object [187]. A digital model of the world and the knowledge of the position (distance, direction and dimension relative to a vehicle) of other objects is a crucial element to enable planning of vehicle movement. The context and the semantics behind a tracked object is particularly important, as different users of the infrastructure have different behaviours, intentions and abilities. A great example for semantic importance are cyclists, that are able to quickly change their orientation and cannot be overtaken without a certain space given due to their high vulnerability. Context on the other hand is crucial to derive correct actions from detected objects. If a vehicle in front is waiting for the traffic-signals to change or parked incorrectly for instance are two completely different scenarios. Sensors used for the overall task of environment perception may vary between manufacturers, but can consist of a mix of LIDAR, radar, ultrasonic sensors and cameras that generate a large amount of data as previously approximated in chapter 3.0.3.

The issue of detecting objects in the world around a moving object is nothing new within the market and long researched for services currently sold in vehicles. Those features include lane-departure warning, blind spot assist or lane-keeping for assisted driving use cases (below level 3). Lane detection and blind-spot assistants for instance are a vital part for both assisted and autonomous driving and explored for several years already [126], [158], [227].

There is two sides to the perception of an environment in terms of input. One being sensing itself and one being the digital representation of measurements / signal inputs. Specific sensors, like cameras or LIDAR, have strengths and weaknesses. Whilst LIDAR for instance delivers information on distances, allowing for 3-dimensional representation and detection of objects, it is not immune to noise and does not return any information on colour. Cameras on the other hand do deliver colourful output, but do not deliver distances (without merging and combining multiple camera feeds) and can become obsolete under bad weather conditions. Cameras however are a substantial part of environment perception and the only sensor for perception onboard of current Tesla (Model 3 and Y) vehicles [56].

Plenty of algorithms were introduced over the past years to identify features and objects within images. State-vector-machines and edge detection algorithms (through drastic pixel colour change for instance) are prominent fits for such. Those methods are hand-tuned to detect specific objects in an environment, but lack the option of multi-feature detection [299]. Simple identification of specific objects (like lane markings) however is not enough for advanced assisted driving (level 2 and beyond) creating a demand for advanced detection algorithms.

Neural networks have emerged as exceptionally suited for facilitating the extraction of contextual, high-level, and in-depth features from images in real-time. This advancement is largely attributable to the evolution of both software tools and hardware capabilities [299]. A diverse array of approaches and architectures has been introduced, aiming to accurately identify objects within video frames, each varying in terms of precision and computational demands [115]. A key objective for applications in autonomous driving is the "speed" of analysing a frame or scene and detecting objects in it, as "real-time" object detection is key to enable integration into moving vehicles. Higher system latency could result in detection of a pedestrian on the road taking to long for the vehicle to initiate protective actions.

The approach of using a recurrent Convolutional Neural Networks (CNN) architecture more deeply explored later for RGB-video analysis for instance, resulted in a 17FPS detection rate in 2015 (measured on a Tesla K40) [115], [207] for instance. A different approach of framing object detection a regression problem by Redmon et al. was later proposed in 2016 under the name of You Only Look Once (YOLO) [204]. After the first model being improved and released in various versions (newest version being v8) it became one of the most used models for two-dimensional computer vision due to its accuracy and real time capabilities [170], [292]. The YOLO9000 release for instance performed >40 FPS for resolutions below 544x544 pixels (measured on a Geforce GTX Titan X) in 2016 [205]. A more recent test in 2023 using the newer YOLO v8 versions, presents >1000 FPS for a 640x640 pixel image (measured on a RTX 4070 TI) [242]. Whilst these numbers are hardly comparable, due to the fact that they're measured on very different hardware, they outline the performance increase over the last few years. Notably however, both example speeds are measured with pixel-dimensions greatly smaller than those known within currently built-in cameras for environment perception within cars as shown in Table 4.

Full perception of a vehicles environment must move beyond two-dimensional object detection for a complete representation of a vehicles surroundings. Depth can come from either of three possible approaches, as presented by Qian et al. [22], including cameras (image based through stereo-camera setups [123] or two-dimensional approximation [262]), point-clouds of (LIDAR or RADAR) [140], [202] or a fusion of both [8], [141].

Point-clouds from a LIDAR sensor can, similarly to camera pixels, be trained in a neural network to detect objects. Approaches to enable perception of point-clouds include the transformation of points to voxels (data format to enclose position in a 3-dimensional space), a simplification into a 2-dimensional space or a direct training on the raw point-cloud of the sensor [140]. Voxel based neural networks are currently considered to be a state of the art solution to detecting 3-dimensional objects with their depth. A sample model proposed in 2017 by Zhou and Tuzel [303], that enables 3-dimensional detection from voxels was measured to deliver 2017 <4 FPS (225 ms total latency) measured on a Titan X GPU. Point-based models like PointPillars following a different approach promise greater performance of theoretical 104 FPS [133] and a measured 42 FPS [65].

Presented (high-level) approaches offer object detection for a specific sensor or sensor type (*e.g.* VoxelNet for LIDAR data). An open question however relies in merging different data from sensors into an overall environment model around the vehicle resulting in the question of sensor fusion. The level of merging data influences the abstraction level of such and the ability to possibly derive benefits from raw-sensor-data [285]. In a high-level fusion architecture, that could directly make use of already known models, each sensor would deliver a list of tracked objects, that can afterwards be combined in a central information processing device (to leave it as general as possible) [6]. This approach of merging data on a high abstraction level has several benefits. For once a neural network can be trained on a specific type of sensor data (like point clouds our pixels), making the model more accurate for this specific type of data. On the other hand, this also reduces the size of each neural network creating a benefit in both computational demand and effort. This method also enables physically distributed computation without raw sensor data flowing together in a single node. Fusion on late stage using already labelled data may mean a reduced overall prediction accuracy and a challenge for training of the specific fusion algorithm used [288]. The opposite to the idea of merging labelled data from various sensors is the concept of merging raw data of each sensor in a single point of data aggregation and perform detection on the combined data. Whilst this approach may need more computational power, more precise sensor calibration and may show potential data redundancy, raw data fusion may enable greater accuracy in tracked objects and reduce system latency [288].

Merging data on an early stage however presents several challenges to keep in mind for fusing raw information from sensors of different types. Those include [203]:

- Data modality and the type of measurements
- Conflicting
- Imprecision and data imperfection
- Noise in some sensor data

The data modality and the specific types of measurements, that are to be fused for a specific perception model is a crucial topic. Relying on a single type of sensor, like image data, reduces the efforts of enabling data fusion and makes application of perception models simpler [8]. This is due to the fact, that the representation of an image does differ from the representation of a returned signal from LIDAR or RADAR for instance. Noteworthy as well, is the fact, that a model or architecture made to use the combined input from multiple type of sensors (*e.g.* LIDAR and camera) must be resilient to one of both delivering unusable data. Harsh sunlight facing a cameras sensor for instance limits the ability to use camera input, whilst extreme weather situations might introduce too much noise into at least some frames of a LIDAR sensor [141].

Several approaches to fusing sensor data for perception are being proposed and researched over the past years [288]. Paper [141] categorised concepts and their fusion methodology into the three categories of point-level, feature-level and birds-eye-view fusion. Point-level and feature-level approaches distinguish themselves through the order of introducing a sensor into the object-detection and segmentation chain. Whilst point-level methods project features within images onto a point cloud, feature-level processes project LIDAR features onto an image. Both concepts lead to enabling detection of an object in a 3-dimensional space relative to the sensor used for feature detection. This for instance would be each camera-view for point-level fusion or the LIDAR generated point-cloud for feature-level fusion. Bird-Eye-View(BEV)-level fusion on the other hand refers to the concept of fusing sensor data in a 3-dimensional space relative to the vehicle to perform detection on a combined set of all data in a global space. This also enables the opportunity to integrate cooperative perception through V2X communication, where active elements of the same infrastructure can share their own sensor data for greater situational awareness (*e.g.* a static sensor like a camera positioned on a traffic light) [39], [290] or the inclusion of high-definition maps directly into the perception model [48]. Noteworthy though, this categorisation is non-complete with further concepts that make use of partial early data merging like the proposal of the TransFuser architecture [48].

Another approach, that is currently gaining interest is the concept of training end-to-end neural networks. This is within active exploration or usage at OEMs like Tesla, Volvo and Mercedes the proposition is to create a network, that is able to handle multi-modality sensing (*e.g.* camera, LIDAR, ...) and deliver vehicle action recommendations as an output [68], [105], [136], [171]. While this approach reduces the insight into underlying decision-making and opportunity to influence such on top of the detection-layer of a system, it is envisioned to enable general-use-case autonomy.

Due to a large amount of proposed solutions for each of the three approaches to perception and sensor fusion [39], [141], [235], [288], [290], I continue by identifying common concepts within proposed solutions. This enables to derive information on the overall algorithms and mathematical operations used and marks the basis to extract requirements for hardware. An overview of possible solutions for merging different sensor modalities and their underlying architecture is gained by studying papers and known solutions. These are summarised in Table 5 below and advances the overview in article [288].

Table 5: Overview of solutions for perception with fusion

Solution	Backbone Type	References
YOLO	CNN	[96], [125], [147], [276]
VoxelNet	CNN	[234], [287], [303]
Other SSD	CNN	[150], [169], [297]
PointNet	CNN	[286]
Region-Based	R-CNN, Faster R-CNN	[84], [140], [207]
Transformer Network	Transformer Model	[48], [235]
State Vector Machines	SVM	[296]
End-To-End Systems	CNN, DNN	[68], [105], [136], [171]
Other NN architectures	CLNN, DNN	[23], [123]

The Table above underlines the current state of research in which several approaches are examined and compared to each other. Each approach has its limitations and possibilities, that heavily depend on chosen approach to sensor modalities possibilities of fine-tuning and the general model size. As different models assume different SLR and target varying Operational Design Domain (ODD) for their approach, like detection speed, a direct comparability of model accuracy and performance is difficult. A major common ground however is the fact, that most approaches heavily utilise a neural network of some sort. Convolutional Neural Networks or derivatives of those build a ground foundation for mentioned approaches (see Table 5), that are almost all built on a basic CNN architecture [147]. Transformer networks are the second major opportunity and will further be explored as well.

Convolutional neural networks for perception

CNNs are well discussed and are a known approach for use cases within computer vision or the detection of objects within images [47]. They are a commonly used as part of models previously mentioned, when discussing non-fusion perception models like YOLO [205], [292] or VoxelNet [303] and lay the foundation for larger end-to-end trained networks [105]. A CNN consists of several layers as shown in Figure 8 and consumes an image of some sort as input. "Some sort" in this case describing the values per pixel, that can be interpreted in various ways like a specific colour-channel (RGB) or a grey-scale. Increasing image quality, in terms of resolution or channels, therefore have a direct impact on the complexity of the convolutional model and the processing involved [216]. The convolution layers to follow serve as a possibility to apply filters to the input image with the goal of enabling favourable feature detection [60]. This is achieved by applying a kernel- or filter-matrix to the input pixel-matrix. Within each convolution, a specific kernel matrix (size being a design decision commonly varying from 11x11 dimensions down to 3x3) is applied to the input image. The resulting matrix y is calculated through element wise multiplication and addition of each element in input matrix x and kernel h.

$$y[m,n] = x[m,n] * h[m,n] = \sum_{j=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} x[i,j] \cdot h[m-i, n-j]$$

Convolution layers can be expressed as a series of MVM by applying the concept of Toeplitz matrix conversion [243].

The pooling layer plays a pivotal role within a Convolutional Neural Networks, as it reduces the spatial dimension of the matrix resulting from the convolution before. This reduction in size, achieved through the pooling process enhances the performance by decreasing the computational load. Figure 8 illustrates an example, in which two pooling layers are applied after a convolution each. This technique involves reducing a 2x2 section of the input matrix into a single output element. The calculation of this value is determined by a specific pooling function; Options are using the maximum (max pooling as seen in Figure 8), the minimum (min pooling), or the average (average pooling) of the four input values in the 2x2 matrix.

The repetition of convolution and pooling within the CNN architecture enables progressive refinement of data for enhanced feature extraction. This is steerable through conscious selection of kernel-sizes, the respective filter itself (*i.e.* a specific filter for edge detection [295]) and a definition of the amount of convolutions and poolings. This layered approach enables both better detection rates through filtering and increasing performance of later stages of the architecture.

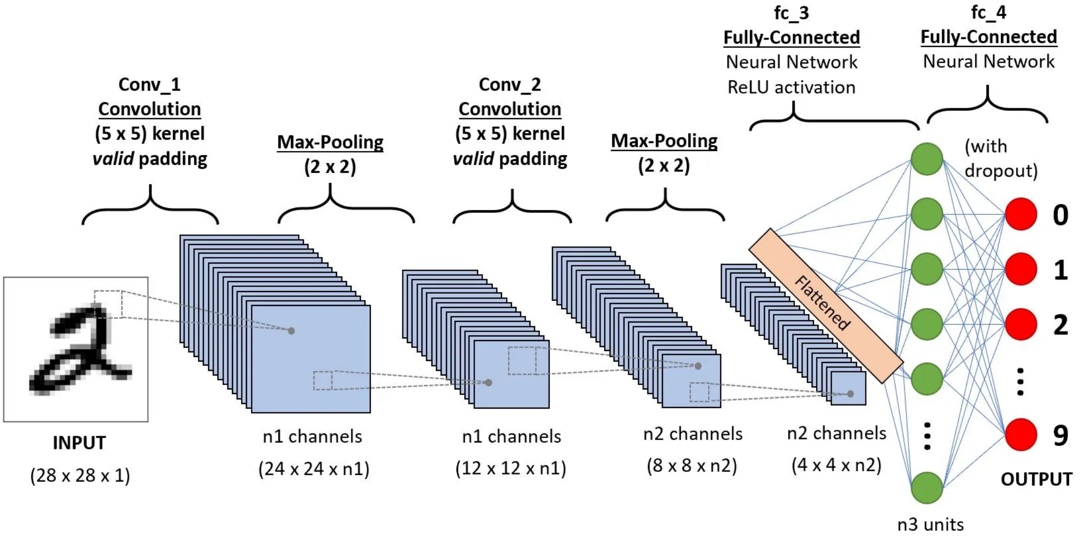


Figure 8: Basic structure of a CNN by [216]
Sample CNN used for detection of handwritten numbers, that makes use of two convolutions and poolings before passing a flattened layer into a fully connected neural network.

In order to feed neural networks' input layer with the result of a convolutional layer-system, multi-dimensional matrices should be transformed into a one-dimensional array first. This is achievable by flattening the last two-dimensional matrix. A fully connected (dense) network being a benchmark in neural network design consists of:

- input layer
- at least one hidden layer
- output layer

Each layer is composed of neurons connected across different layers [216]. A neuron computes an output value by applying an activation function to the weighted sum of inputs from neurons in the preceding layer, along with a bias value. Weights and biases are pre-computed values derived from model-training in advance. Input values are propagated from upstream layers of the network [213]. The underlying mathematical operation within a single neuron is expressed as [217]:

$$\mathbf{A} = f(\mathbf{WX} + \mathbf{b}) \tag{1}$$

Where:

- WX represents the matrix-vector multiplication, producing a vector where each element is the weighted sum for a neuron
- b is added to each element of the resulting vector from WX aligning with vector addition.
- $f(x)$ representing the activation function

The dimension of matrix W and vector X depend on the amount of fully connected neurons within previous layers of the NN.

The last hidden layer propagates its result to the output layer of the network, in which all inputs are once again used to compute a final numerical value. The output of neurons on the output layer however correspond the a probability of a specific class. The neuron with the highest likelihood indicates the networks overall prediction. A certain neuron for instance could be interpreted as the "car"-neuron, meaning that with a high probability this neuron would label the input data as a "car".

A Convolutional Neural Networks, as described above, is considered a crucial part of most advanced perception models, like YOLO or VoxelNet. The amount of layers, size of matrices and the specific pooling or convolution used, is subject to a specific use case and is impossible to generally define. As discussed, CNNs are not the only part of the overall perception model, but are part of the backbone architecture used within larger models. Applications include region-based CNN (R-CNN), Fast R-CNN, Faster R-CNN, single shot multi-box detector (SSD) and YOLO [147]. I will therefore briefly explore the architecture of the approaches previously identified within Table 5, to increase the understanding on more specific algorithmic traits.

Previously discussed YOLO framework is a commonly used model for object detection and is known for its real-time application and the fact, that it utilises a single neural network for bounding box prediction rather than a mix of networks trained for a specific feature class [170]. YOLO has been published in several versions over the past years and has since then varied in its underlying architecture. Current versions of the model primarily use a CSPDarkNet53 backbone, which is a convolutional network with 29 layers and a kernel size of 3x3 [30], [118], [160]. The YOLO framework also supports manual switching of the backbone to other models like ResNet, EfficientNet or VGG [160], [198]. All three options are similar in terms of the overall architecture, but use a different amount of convolutions and sizes for poolings, resulting in a change in both model performance and detection precision. The ResNet framework exists in several sizes, but is known for democratising skip-connections and residual learning [99]. Those two elements are interesting in the context of chip-design as well, as they reuse a result of a convolution-step x as an input in another convolution-step $x + 2$, making a fast access to memory necessary for high performance, thus making specialised caching interesting. The ResNet 50 architecture (being a comparable size to CSPDarknet53) features a total of 33 convolution layers with a kernel size of 3x3 leading (first convolution uses 7x7 kernel) into fully connected NN with 1000 neurons [99]. The VGG model in comparison features a smaller amount of just 16 convolutions (3x3 kernel size), but utilises a total of 3 fully connected layers with 4096 neurons (2 layers) or 1000 neurons (1 layer) [99], [233]. Lastly, EfficientNet incorporates several opportunities of scaling in both model-depth and width to increase performance, but relies on the same base architecture being a mix of convolutions (3x3 and 5x5 kernels) and 17 convolutional layers in its first implementation [246], [248]. Further "neck" and "head" elements of the YOLO architecture are needed for generating a final output or connecting parts of the overall model. They primarily consist of further convolutions and feature pyramid networks (closing the gap on different resolution levels through scaling and addition) [118], [145].

Voxelnet is based on a modified architecture, when comparing it to previously summarised convolutional networks. Instead, it is based on a region proposal network, that is created on the base of a region based convolutional network (R-CNN) [65]. Additionally to a CNN, a R-CNN extracts region proposals, performs feature detection on each and merges features from various regions to classify regions within the image [84].

PointNet is another option for handling three-dimensional point clouds. The underlying architecture is a deep neural network, that consumes raw data instead of relying on voxelization for further work making it stand out from VoxelNet. A network consisting of input transformation, feature transformation, point-feature aggregation and a segmentation network relying on matrix multiplications of up to 1024 [196].

Single-Shot-Detection refers to the concept of performing object localisation and classification in a single pass through the model, instead of proposing regions first and performing feature extraction afterwards, as does VoxelNet. YOLO, as described in detail above, is a sample model for the kind of single-shot detectors. Other architectures may utilise other backbones are models for detection, but are similar in their overall concept of using convolution, pooling and a fully connected network [150].

Transformer models for perception

An alternative to already discussed CNN (in some way or another for R-CNN, VoxelNet, Yolo), is the idea of using a transformer architecture, which is known for being a newer and more promising approach [166]. Well used for building language models, transformer models have shown to be more efficient in training and resulting performance [266]. The concept of incorporating an attention mechanism enables modelling of dependencies throughout a large number of input sequences. Figure 9 shows a basic structure of a transformer model, that consists of an encoder (left) and decoder (right), which are each composed of a stack of identical layers (N). The model's main characteristic is the Multi-Head-Attention block (Figure 9), that consists of several matrix-vector, matrix-matrix multiplications and scaling operations (proposed size of 64x64) [266]. Another key-element of the attention model is the fact, that each attention uses the previous attention as an input on top to current-attention input (making use of a feedback mechanism), as shown in Figure 9. This enables the network to do better predictions by incorporating older steps impacting the current one. A great example for

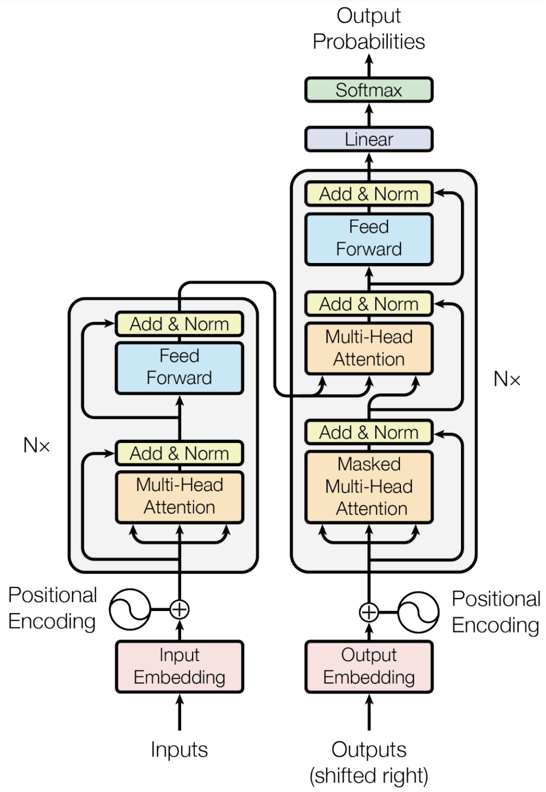


Figure 9: Basic structure of a Transformer model by [266]
Transformer models make use attention models, that are part of 2 elements. The first element uses a new input for processing whilst the second part receives the previous result as input.

this is the generation of a sentence, in which words previous to the current one have an impact on the next one. Using `you X` as an input for instance leaves open a great amount of possible next words, whilst `How are you X` suggests a specific type of word (X) to follow next.

Transformers are now well known for their application in language processing and their benefits over CNN architectures. They enable building dependencies over input and output elements, allow for massive parallelism and require minimal inductive biases. Transformer networks are also able to process multiple modalities using a similar architecture - thus eventually enabling detection in point-net-spaces and pixel-spaces for 3 and 2 dimensional perception [124].

Aside from the great impact on language processing enabled by using transformer models, it is also an opportunity to employ this kind of architecture to computer vision. An idea proposed by Dosovitskiy et. al. is to feed a 2-dimensional image into a standard transformer model by flattening the image into a single dimension; comparable to reducing the dimension of the last convolution or pooling layer of a CNN to feed it into the fully connected neural network [72], [216]. This approach of flattening an input is further supported by other papers, like the introduction of the TransFusion framework [48], that makes use of earlier mentioned intramodality of sensors and combines LIDAR and camera sensing into a single perception framework.

Comparison of approaches

Reviewed paper landscape shows, how CNNs are seen as a state of the art solution for environment perception with ongoing challenges in combining different modalities or sensor viewing angles. Using transformer models for computer vision tasks embodies a newer approach, that promises smaller training cost and a simpler integration of varying sensor types. Both approaches are a concept that can be modified to specific perception task or requirements (precision or detection time), by scaling the model in its width or depth [48].

Analysing a commonly referred benchmark platform enables further understanding of the relevance and effectiveness of previously described architectures. Close to the presented use case of environment perception, of identifying objects on the street network, lies the KITTI dataset (presented in [87]), often referred to in research articles [125], [141], [228], [288], [303]. The KITTI dataset provided by the Karlsruhe Institute of Technology (KIT) consists of real-world data captured within and around the city of Karlsruhe. A set of cameras (stereo, monochrome and RGB) and a 360° laserscanner were used to collect data on a total trip length of 39.2km. The resulting dataset is known to be close to reality being able to represent a close to realism benchmark for newly proposed solutions for environment perception in the automotive market [88]. Two metrics of the benchmarking suite are particularly of interest: The processing power needed per inference and the accuracy of the model.

By examining the leaderboard (status January 2024) of the KITTI benchmark [129] and assessing the architectures behind best performing models, one can achieve an overview of the role of either

CNN or transformer models for "best performing" environment perception. Notably however; the KITTI dataset is one possible opportunity for benchmarking and may, due to its nature of data collection in Germany, not be representative for traffic situations or other infrastructure users in other regions such as China or India. Table 6 below depicts a list of best performing models in the KITTI 3-dimensional benchmark for car detection, that has a total of 416 entries. The list is non complete and models without documentation (no paper, code or publication attached) are left out, as they cannot be analysed for the specific architecture proposed. Performance metrics not generated on the benchmark hardware (1 core clocking at 2.5 GHz), are also left out due to its in comparability in terms of runtime per inference.

Table 6: Overview of best KITTI 3D-car benchmark performers [129]

Solution	Accuracy (%)	Runtime (s)	Architecture
VirConv (-S, -T)[1]	87.20 (86.25 -T)	0.09	CNN [284]
UDeerPEP[1]	86.72	0.1	CNN&Transformer [70]
HPC-NET[1]	85.50	0.18	CNN [298]
PVFusion[2]	85.20	0.01	FPN&Transformer [274]
LoGoNet[2] [3]	85.20	0.1	CNN&Transformer [139]
CasA++ [2] [3]	84.04	0.1	CNN&Transformer[283]
OcTr[2]	82.64	0.06	Transformer [301]

[1] Model listed in position of overall best performers in terms of accuracy

[2] Model listed in correct ordering but not correct position to showcase runtime difference

[3] Model is also amongst the Top-10 for Pedestrian and Cyclist detection

Table 6 greatly shows, how the two previously discussed approaches to perception only minimally distinguish each other in precision within the 3D-car-benchmark. The pure transformer-based model OcTr stands out with a worse accuracy in comparison to outperforming other CNN-based architectures in runtime. The Table further depicts how several approaches can be used to create comparable performance in both runtime and accuracy, making a decision for a certain technology difficult for these two metrics only. Best performing models however do include a mixture of CNN and transformers. Notably, two models using transformers are also amongst the best performers of pedestrian and cyclist detection. With the "perfect" algorithmic solution to environment perception still being a question of research, it is difficult to name a specific architecture to implement. Current research and benchmarks (see Table 5 and 6) suggest that future models could be a combination of transformers, CNN and possible additional components (FPN, fusion layers, ...). This makes it necessary for a proposed accelerator chip, to be multiverse and not limited to a specific framework.

Additionally, the specific level of fusion is a very individual matter that depends on the sensor type used and the necessary accuracy for environment perception. Tesla for instance is known to have dropped ultrasonic sensors from their vehicles and solely relies on a camera setup for near-field observation. When questioned about the reasons for this elimination of sensors, the former Head of Tesla AI cited the complex sensor fusion of two very different types of sensors, which also increased the effort of calibration [253]. The fusion-stage and the localisation of such does have an impact on processing power needed at a central stage. A late-fusion design could make use of distributed computing and employ LIDAR sensors that each perform a detection and serve a feed of bounding boxes. An early fusion design on the other hand, would use raw-input data from each sensor for a central perception model. An abstraction level in between, where raw LIDAR data for instance is cleaned within the sensor itself, is possible.

Impact of Noise

In the exploration of algorithms and models pivotal to advancements in computer vision, a critical inquiry persists regarding the influence of noise afflicted hardware for the use within neural networks. The potential degradation in model accuracy, precipitated by the integration of photonic accelerators, warrants rigorous examination. This concern is particularly pronounced given the high importance of reliable object detection for safety-critical applications. To deeply understand the ramifications and potential benefits of incorporating photonic components, an extensive series of simulations, coupled with free-space experimental investigations have been undertaken. These efforts aim to provide a comprehensive understanding of the impact of photonic technology on the performance and reliability of computer vision systems.

Ong et. al. serve a possibility for comparison through an experiment of integrating an MZI array for a CNN [179]. The test uses a CNN for image detection on the MNIST dataset, which is a test for image detection of hand-written numbers in an image of 28x28 pixels [66]. Whilst not comparable to the complex situations of environment perception in real world traffic scenarios, it outlines the impact of noise on convolutional networks. The performance test used for reference in [179] achieves an accuracy of 99.6% on a purely electric hardware. A photonic model simulation yields an accuracy of 99.2% being a decrease of a relatively small 0.4%. Other papers following the same concept of a simulated MZI based accelerator show a larger impact of noise, that reduce the overall model accuracy by 1% [64], [117] or 2% [185]. Experiments in free space utilising the concept of Multiplane Light Conversion produce similar results. A relatively small decrease (<4%) in model accuracy, that could further more be reduced by implementing the concept in a more controlled environment (*i.e.* an integrated circuit). [33], [146].

Summary

This chapter analysed the complexity of environmental perception necessary for assisted or automated driving, elucidating several critical aspects. A persisting open question is the determination of a minimal sensor setup; encompassing their sensor type (modality), the volume of data they generate, and their integration into possible sensor fusion essential for achieving Level 4 automation in driving. This level of automation necessitates a vehicle's capability to perceive its environment sufficiently to operate without human intervention under specific conditions, as previously stated. It is acknowledged that processing large datasets requires significant computational power and a sophisticated level of data fusion, if not using an end-to-end trained network.

The multimodality of sensors presents a formidable challenge in environmental perception and the associated software methodologies, yet it is deemed indispensable for ensuring system resilience in the face of variable weather conditions and road scenarios. Current research efforts are focused on devising efficient and precise methods for the detection and segmentation of other infrastructure users, with transformer architectures and Convolutional Neural Networks being prominent contenders for this task. Both concepts serve as fundamental frameworks, extensively employing matrix operations and data propagation techniques, whether through the sequential layering of attention mechanisms in transformers or the interconnection of various convolutional layers.

An open consideration in making use of photonic accelerators for computer vision tasks, is the potential degradation of model accuracy attributed to hardware induced noise. Nonetheless, the prospective benefits of photonics, such as a reduction in energy consumption and the decrease in heat production, present a chance for its application within vehicle processing.

4.1.2 Video Rendering

Rendering graphical content is another use case standing out due, to its requirement of real-time latency and high throughput for advanced scenes. Specific use cases can be two- or three-dimensional, depending on the specific customer service. Sample graphical content relies within animations on the infotainment screens or animated vivid representations of personal assistants. Whilst animations could potentially be pre-recorded and played back when needed (if not specific to vehicle context), virtual assistants need to be rendered to a specific text spoken at any time and cannot be prerecorded. This last example is dependent on the visual representation of the avatar and its complexity. Current BMW's for instance, feature an abstract shape floating on a specific screen [194], that stems little complexity. NIO as another example is using an abstract "smiley" based character called Nomi with a limited amount of specific gestures [173]. An example of a complex avatar, that must be rendered on the fly, is MINIs newly introduced Spike avatar - a dog communicating with the user [193].

Three-dimensional graphical content (Augmented Reality (AR), Virtual Reality (VR)) has already been introduced within latest vehicles across several OEMs and include AR content for navigation and drivers assistance systems as seen in Figure 10. OEM publications, conferences and trade-fairs give a glimpse of a trend, in which the automotive industry is increasing its development within advanced entertainment use cases with virtual, three-dimensional environments [20], [94].

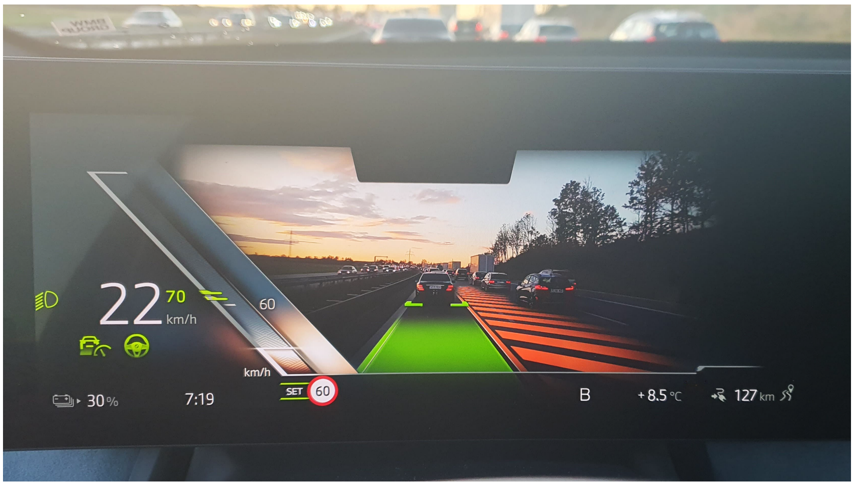


Figure 10: BMW iX Traffic-Jam Assistant in AR
Modern vehicles may be equipped with AR content display, as shown within this image. Here shown use case is a visualisation of assisted driving in traffic jam situations. Its highlighting tracked vehicle, lane and freed emergency service lane to enable a drivers understanding of vehicle actions.

Building a scene in AR or Virtual Reality (VR), or Extended Reality (XR) as clustering name, consists of a variety of steps that range from modelling three-dimensional objects to the transformation and lighting of objects for a specific two-dimensional output medium. The steps between a list of polygons and a definition of specific pixels on screen can be summarised as a rendering pipeline, being especially critical in the context of XR, as it not only ensures realistic visual representation, but also enables the real time interaction necessary for user immersion.

Key elements of a rendering pipeline include rasterization of polygons onto pixels, transformation between different object-spaces, computation of light and colour for each pixel [268]. All of these functions rely on matrices for mutation of polygon-corner-points and pixels, which can be represented as a vertex [131]. Each value within those vertexes or matrices can be represented by different bit-depths dependent on their use case. Whilst the standard representation of a vector in OpenGL for instance uses a float (32-bit floating point value) [271], specific applications may reduce the precision to 8-bits (short) for an increase of performance [215]. The specific workload and the amount of computation power needed for rendering a specific scene is dependent various factors. With the amount of light sources and a desired level of detail in shading set, remaining two factors are pixel-amount and object complexity. The amount of pixels linearly impacts the amount of computation needed as shading must happen for each pixel. The amount of objects within a scene and its level of detail is impacting computational workload exponentially [165].

Two-dimensional representations on vehicle-screens are bound by the display resolution they are shown upon. Display resolutions, that are a basis for approximating rendering expense, vary between OEM and models specifically. Example for high-resolution displays are the BMW theatre screen (7680 × 2160 pixels) [192] or the Mercedes Benz MBUX Hyperscreen (3,088 × 1,728 pixels) [252]. Additional examples are the Tesla 16:9 main screen (3000 x 2000 pixels) [111] or Central Information Display (CID) within the Ford F150 (1200 x 1920 pixels). With the exception of gaming services or video-streams (possibly more) specific rendered content will most likely not fill the screen [194] [252] but will be limited to certain parts of the total display. The rendering pipeline for 2-dimensional scenes specifically consists of various MVM of a 4x4 Matrix with a 4x1 vector for transformation of fragments and vertices and rasterization [62].

Use cases in XR increase the need for rendering a world beyond the actual screen size and the currently seen objects, to enable moving your head without any display lag or loss of quality caused by eventual re-rendering of content. Displays within XR devices are in a similar resolution range to displays mentioned before with examples being the Quest 3 (2,064x2,208 pixels) [109] or the just released Apple Vision Pro (3,648x3,144 pixels) [172]. Indulging in virtual words further workload beyond positioning, as positioning the viewing device within a three-dimensional room is crucial for the experience. This is done by implementing a Simultaneous Localization and Mapping (SLAM) method to localise and map existing objects in the real world to references in a virtual word [190]. Just like the mentioned rendering pipeline for two-dimensional display feeds, the SLAM functionality heavily relies on matrix (4x4), vector (4x1) multiplication, due to its simplicity and efficiency [85]. Bit-level precision of each data point is not standardised and may vary between different sensing setups. Nvidia's Deep Learning Dataset Synthesizer for instance states a maximum precision of 16-bit for generated scenes [210], whilst Amazon Science presents a performance increase by adapting their SLAM methods from 32 bit-floats to a precision of 16 bit float [197]. With little information found on 8 bit SLAM (apart from thermal applications), this leads to the assumption of a minimum of 16bit precision for localisation and tracking in AR use cases.

To summarise: With presented data at hand an optical accelerator for graphic use cases would need to feature a bit-precision of at least 32-bit (for transformation processes) and a matrix dimension of 4x4. Matching this with photonic hardware opportunities presented in Section 2.1.1, leads to the observation, that to accelerate currently existing graphical pipelines one would have to implement a CrossBar array. Neither MPLC, MZI nor WDM offer the bit-precision required.

An open question however relies in the impact of noise in optical accelerators onto rendered content. I will use the standard 2-dimensional OpenGL rendering pipeline and analyse the specific algorithms used and the implications of using a noise affected computation. I will first analyse the mathematical operation of a transformation itself giving an outlook on the possible implications to a rendered object and then simulate noise in a rendering program producing objects of different size. It must be noted however, that noise is not equally distributed over the whole signal range Instead it is measured to be normally distributed, as seen with experimental implementations of previously mentioned optical integrations [149], [168]. Due to the fact, that accurate tests of noise distribution a 4x4 MVM-accelerator based on the crossbar array architecture are none existent, I will assume 99% numerical precision across the overall frequency range and therefore across the entire spectrum of MVM.

Transformation within Rendering Pipelines

Transformations of objects are a vital part of rendering and mutating content on screen. Those transformations include scale, rotation, shearing or simply moving an object within a certain space. Given a transformation matrix A and a translation vector t, the operation of translating a point in a 2-dimensional space can be written as:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = A \begin{pmatrix} x \\ y \end{pmatrix} + t$$

where $A = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and $t = \begin{pmatrix} t_x \\ t_y \end{pmatrix}$ are the parameters for a simple shift in both x and y direction defined by the t_x, t_y.

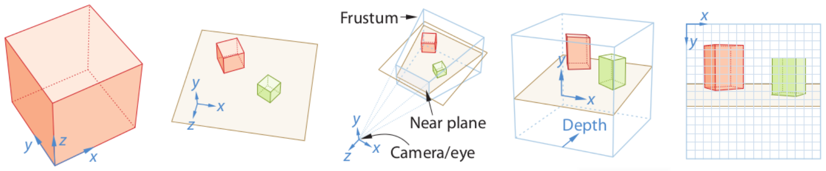
This transformation is sensitive to noise in both matrix-vector multiplication and addition. This results in not accurate transformations of objects affecting position, orientation and shape of objects within the world. This can easily be demonstrated by an example calculation.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 25 \\ 70 \end{pmatrix} + \begin{pmatrix} 50 \\ 50 \end{pmatrix} \overset{1\%\text{noise}}{=} \begin{pmatrix} 74.0025 \leq x' \leq 76.0025 \\ 118.107 \leq y' \leq 121.907 \end{pmatrix}$$

The formula above shows, how the result of the computation is not the expected numerical result of $\begin{pmatrix} 75 \\ 120 \end{pmatrix}$ but anything in between a certain range dependent on the actual noise of the system.

Testing the impact noise to rendering pipelines can be done by simulating noise within a state of the art rendering pipeline. This consists of various transformations and optional vertex and fragment shaders, enabling further manipulation of objects. A vertex shader is a piece of software executed for each vertex that is a combination of position and optional data (*i.e.* colour or texture coordinates) for each corner of a fragment. A fragment is the simplest two-dimensional object, a triangle, that is defined through 3 corners (vertices) each. A fragment shader respectively is a piece of software executed for each fragment and the pixels associated to it and is executed after the vertex shader.

A mandatory step in building complex scenes, is the transformation of objects (simple coordinates, triangles, ...) into respective spaces needed for further use. Those, shown in Figure 11 consist of the model-space in which a certain object is built and modulated in, a world-space to collectively arrange all imported objects into a uniform 3-dimensional system, camera oriented space (and a normalised version of it) and the actual screen, being a 2-dimensional system. Transformation is based on mutating points (vertices) through the defining a transformation matrix of each space. This is done mathematically through MVM of a 4x4 matrix and a 4x1 vector being a specific vertex with an added fourth element making it a homogeneous coordinate needed for correct transformation.



(a) Model (b) World (c) Camera (d) NDC (e) Screen

Figure 11: Spaces within rendering pipelines (OpenGL)
Here shown five spaces give an insight into the five transformations needed for objects within a rendering pipeline. Modelspace (a) refers to the individual space for each object. Worldspace (b) refers to a global space to place individual objects within. Cameraspace (c), Normalised Device Coordinates (NDC) (d) and screenspace (e) refer to intermediate steps towards displaying objects on screen and rasterize them later on. Image is taken from a Computer-Graphics lecture at OTH Regensburg held by Prof. Dr. Kai Selgrad

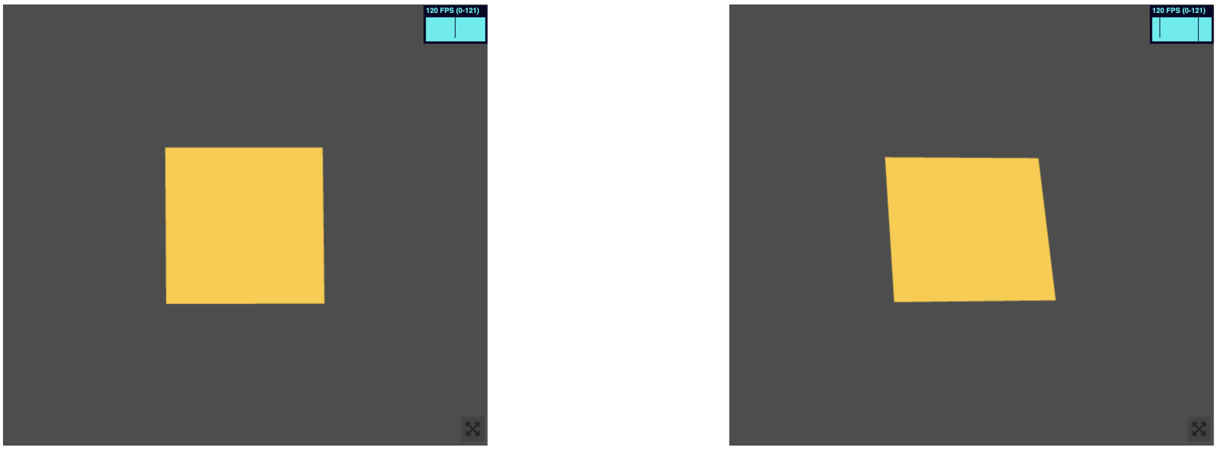
The pipeline shown in Figure 11 can quite easily be integrated within a small program using OpenGL Shading Language (GLSL) and an interpreter. I build a small sample, that outputs a simple yellow square. This is realised by implementing one fragment-shader, one vertex-shader and a "main"-file for the overall logic needed. The transformation depicted in Figure 11 is implemented within the vertex-shader, that receives all uniform matrices and the vertex position as input.

```
    gl_Position = projection * view * model * position;
```

This single command performs 3 MVM and returns the position (4x1 vector) as an output for the further process of rendering (*i.e.* in fragment shader(s)). Depicted spaces and the transformation amongst them, addresses an important factor in evaluating potential benefits of optical accelerators as it contributes to the question of data loading and therefore, affecting setup times and the speedup gained. Whilst the transformation from model to world-space is executed per model, meaning that the transformation matrix is dependent on the specific model, all other transformations are uniform amongst all elements within the world space. This means that the elements of the transformation can be loaded once for a larger amount of computations greatly reducing the setup time and enabling high-performance batch processing.

To simulate this, I modify a working GLSL program within WebGL [32], that displays a simple square and extends this by simulating noise. The original version, without random error renders the yellow square depicted in Figure 12a. Noise can then be simulated by adding a random error to each of these multiplications. The book of shaders proposes using a combination of a fraction and the sine function $y = fract(sin(x) * 10000.0)$ for creating a seemingly random number within a shader [270]. This can be used to build a noise function, that adds a random error factor to a specific value with a certain upper bound. In this case $1 - crossbarprecision$ accounting to 1%

potential error, using noise approximations as previously explored in Section 2.1.3. Rendering the same object from before with an adapted calculation of the vertex position `gl_position` returns an irregular rectangle instead of the expected square, as seen in Figure 12b. As noise has only been simulated for the transformation of the object, the colour remains unchanged.



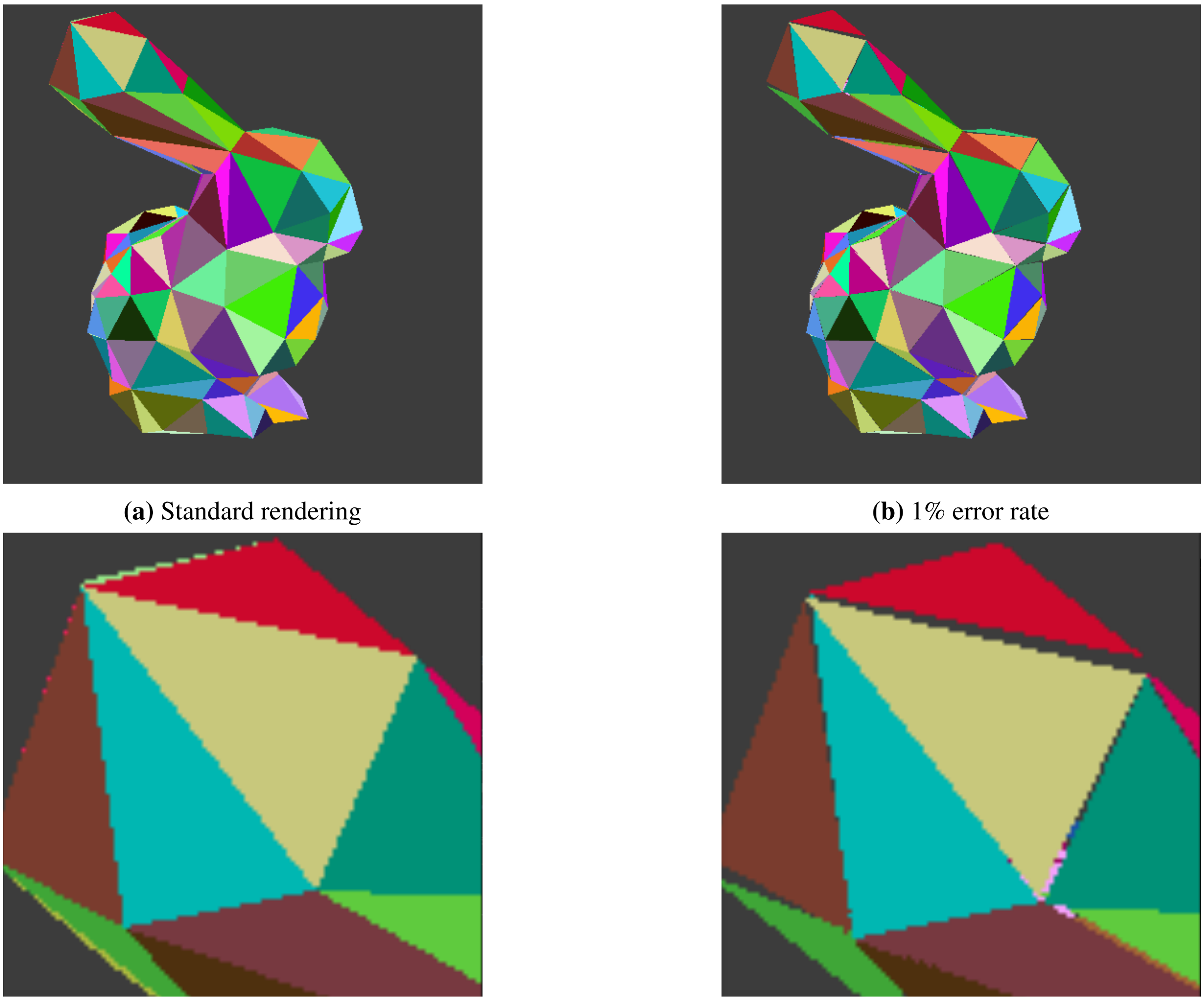
(a) Standard rendering

(b) 1% error rate

Figure 12: Rendering of simple square in GLSL
Simple renderings of yellow squares of equal size. The right version (Figure 12b) is rendered with a simulated noise of 1% within transformation.

The simulation of noise for rendering a simple object is coherent with the expectations from the simple mathematical test above, proving that shape distortion may be an issue. As rendered scenes however usually not consist of a single object, but are likely to consist of several thousand polygons (500.000 polygons for a vehicle in a recent video game for instance [255]), the importance of slight shearing of vertices of a single polygon is questionable and must further be explored through another test. To approximate and get an understanding for the importance of polygon distortion, I simulate rendering of a larger object consisting of a total of 292 polygons representing a 3-dimensional bunny.

I will use OpenGL Mathematics, which is a library for C++ based on GLSL due to its simplicity. The model of choice is a list of polygons, derived from the "Stanford bunny" greatly broken down in its complexity. Similar to adding noise to rendering a simple square in Figure 12a, one can simulate error within the OpenGL pipeline. For comparison of both no-error and 1%-error hardware, a bunny is rendered twice with the same settings apart from noise simulation. The result of rendering is a png file shown in Figure 13, displaying the bunny from the side with a random colour for each fragment. Not applying a texture or an equal colour to all rendered triangles is helpful in this case, as it enables an even clearer representation of triangle borders and possible distortion. Figure 13a shows the rendered output without error and Figure 13c depicts a greatly zoomed in part of the no-error bunny rendering. One can clearly see the borders of each polygon, lining up with the ones next to it. Figure 13a shows the rendering result with an added noise of 1%. Whilst it is no subjective measure to compare two images visually, the object (bunny) is still easy to make out and the overall shape looks good. However, whilst harder to spot on the total image, zooming in on the image (like depicted in Figure 13d) shows, how polygons no longer match up in every single instance resulting in a broken object with the background and theoretically obstructed polygons shining through. This leads to the observation, how an increasing size of polygons relative to the renderings display negatively impacts the fault created by the noise in MVM. An observation to be further analysed with more complex scenes and smaller fragments.



(a) Standard rendering

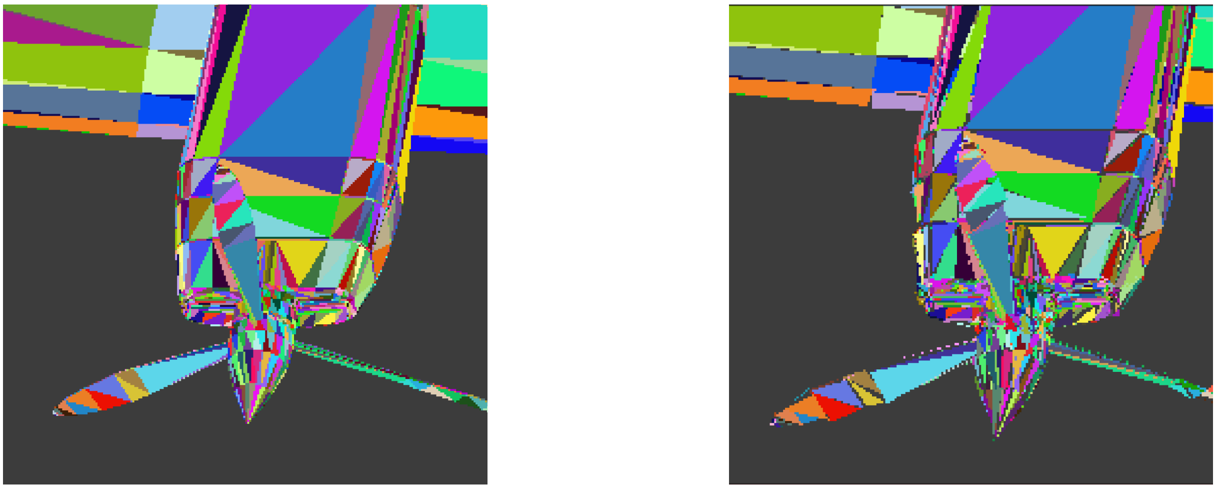
(b) 1% error rate

(c) Standard rendering (500%)

(d) 1% error rate (500%)

Figure 13: Renderings of bunnies with different error rates
Results of an OpenGL rendering pipeline to output a bunny. All fragments of the bunny use a random colour for polygon distinction. Images 13a and 13c show the result of a standard implementation without noise. Images 13b and 13d depict the result of a rendering with 1% random noise

Whilst the effect of distortion and unconnected polygons might not directly be noticeable for a geometry with small polygons (in comparison to the total display size), they are directly apparent at some point, when zooming or scaling a specific object. To further analyse the behaviour of noise and the implications on the appearance of objects, I tests an 3-dimensional model of greater complexity. This is done by converting an `.obj` from John Burkardt [34] into a usable format for OpenGLM and rendering further tests. A Cessna twin engine aircraft modelled consists of 7445 triangles of varying sizes, with the engine being particularly interesting due to its small polygons. The code behind rendering this object, is applied from the previous example of rendering a bunny, with the difference being resolution of the overall object (rendered in 2000x2000 pixels) due to model size. Figure 14 shows a close-up of one of the planes engines depicting a more complex scene (in terms of number of polygons) in comparison to the bunny shown in Figure 13. Whilst the noise-afflicted rendering in Figure 14b shows distortion similar to the bunny above, this example also shows how distortion can become increasingly irrelevant with decreasing size (relative to the screen) of rendered triangles. The shape of the object is clearly identifiable, even with slight errors in the image.



(a) Standard rendering (500%)

(b) 1% error rate (500%)

Figure 14: Cessna renderings (500%)
Renderings of a Cessna 300 engine zoomed in at 500%. Each polygon is randomly coloured. The left image shows the expected result, whilst the right image is simulated to be resulting from noise afflicted processing.

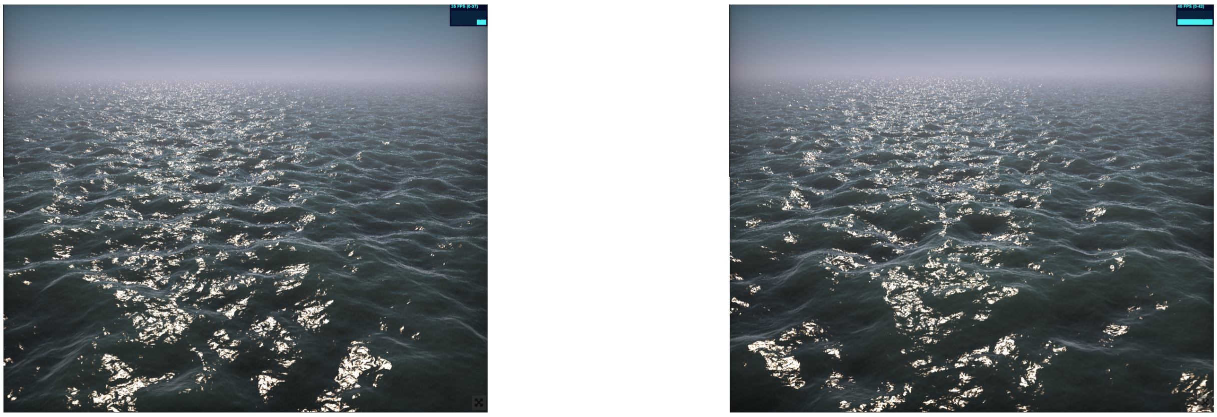
Experiments with object transformation have shown a questionable usability for optical acceleration. The impact of object distortion, can reduce the image quality to an unusable state. Several approaches and techniques however could potentially increase usability of optical, analogue accelerators as discussed with Prof. Dr. Kai Selgrad (Professor at OTH Regensburg for computer graphics) . For instance, one could pre-compute the matrix-multiplication of different transformation matrices across different spaces introduced in Figure 11, to reduce the impact of noise. This is possible due to the fact, that the same transformations are used for all elements of the object after conversion into the world-space. Another possible idea is to include a hybrid approach to the pipeline and render objects out of frame with noisy hardware, but stick to common digital computation for those polygons directly visible to the user.

Shading as Part of Rendering Pipelines

Another vital part of a rendering pipeline however are the shaders used for calculating light and respectively the color of a pixel. Shadows and light (*e.g.* reflection of the sun, brightness of an object space) are crucial to representation a virtual object on a 2-dimensional plane. There are several approaches and concepts to compute lighting of a specific object. Both the perceived realism and and needed computational performance are criteria to choose a specific model for shading [98]. Shaders and lighting models are commonly based on evaluating, if a specific pixel is hit by a beam of light and how much of that light it reflects towards the camera. This is influenced by several parameters, like the shinyness of the material used or the intensity of the light-source. This idea in its simplest form can use a single source of light or evolve into hundreds of sources with light propagating throughout several planes of the object increasing the perceived realism of a rendered scene. The computation involved however, is based on a few general algorithms involved. Those include the calculation of normals, the mutation of vectors (3x1) and the addition of floats [36]. To simulate the effect of hardware noise onto shader programs, I use a sample WebGL project (as seen in Figure 15), that renders a moving wave with light reflections caused by a single source of light. In this case a sun positioned outside of the viewing plane. The sample project uses the Phong Shading model, which computes light as an addition of three types of light. Specular, diffuse and ambient light. Each component can be calculated through multiplication of different vectors. Notably for performance though, different from previously discussed transformations using the same transformation-matrix for a great amount of data, the vectors involved in shading differ from pixel to pixel. This is due to the fact, that for each pixel on the screen, one calculates angles individually. This however implicates that for each computation one must load both the weights and data for every single computation having a large impact on setup times.

```
    light = (diffuse * light) + (spec * specular * SPECULAR_COLOR) +
        ambient;
```

This project can be enhanced with a simulation of noise by adding a noise factor to each mathematical operation, as previously seen in rendering a simple square above. Adding a random error to each multiplication and addition results in the rendered image seen below in Figure 15b. It is noteworthy, that the project uses random noise for the generation of waves itself, making each rendering somewhat unique and hard to compare on a pixel level. The introduced error in computation however yields no visual decrease in quality of the image. This observation makes sense, as the error in brightness of a specific pixel becomes noticeable through a big difference between neighbouring pixels. The error of 1% per pixel is of minimal impact for the absolute difference in pixel brightness and therefore has a relatively low impact on overall image quality. This simulation shows, how shading is quite resilient to hardware induced noise within computation and leads to the assumption, that a photonic hardware accelerator could be of great use for this specific step within the rendering pipeline.



(a) Standard rendering

(b) 1% error rate

Figure 15: Wave renderings (animation)
Screenshots from an animated scene showcasing motion of waves and reflecting light. Movement and position of a wave is generated with a random function. The right image is showing an image with a simulated error of 1% within MVM operations.

Figure 15 shows, how noise has no or at least non noticeable impact on image quality in an existing rendering pipeline, when introducing random noise into the shadering process. Noise may however be considered beneficial to the realism of a rendering and is seen as a key component for graphics rendering systems. It is used to help with creating natural looking textures, improved realism for lighting and simulation of visual effects [132], [220], [239]. This opens the question of the possibility to actively make use of hardware noise for improving visual quality of renderings. An assessment of state of art noise simulators and an experimental test of analogue error introduction as a replacement, may reveal a greater idea of the gains and losses of this idea. Notably, the distribution of error for a certain photonic accelerator must be considered for this experiment, as different implementation of optical technologies, introduced in chapter 2.1.2 are subject to a specific distribution of error (*i.e.* normal distribution as measured in experiment [168]).

The remaining question and the incentive of this Section relies in the usability of optical components for rendering pipelines or the rendering of graphical content in a broader term. An analysis of the amount of pixels and the associated workload suggested a great potential for optical components due to the high quantity of data processed and latency requirements for a seamless user experience. Analysing the algorithms behind graphical rendering processes has shown a great match for already discussed photonic implementation MVM and yields no imminent issues for an implementation with a crossbar array setup on a theoretical basis. The practical use however remains questionable, as the experiment of simulating noise within a current state of the art rendering pipeline has shown. Distortion of polygons shown in Figure 14b might reduce the quality of visuals by so much, that optical components aren't seen as good enough for object transformations. Experiments with scenes of different complexity lead to the assumption, that fragments further away from the camera-position are more resilient to noise. In contrast, simulations on hardware noise within the shadering process have shown, how the impact of numerical error is not visible in actual results, as seen in Figure 15. This does however open up the question, if optical or noise-afflicted hardware in general is suitable to general graphical renderings or if at least certain parts of the rendering process must happen with greater precision to enable a high quality image of large polygons. If only parts (*i.e.* the shaders) of the pipeline are able to run on an optical hardware accelerator, this arises the question of the overall achievable speedup and efficiency gain, as a partial offload to an accelerator increases the effort of managing parallelism and data movement.

As previously noted, issues related to the implications of hardware noise and the resulting distortions in object transformations could potentially be mitigated through post processing or go as far as become a quality-increasing factor for realism in the shader process. An examination of different filtering techniques and their effect on noise-afflicted mathematical operations relies outside of the scope of this thesis. Simulations however have shown, that application of analogue accelerators for transformation processes is troublesome and requires further tests with in-depth processing or a modification of state of the art processes. Whilst this opens up an interesting field for investigation, it makes the photonic acceleration of graphic-rendering-pipelines a greater effort of hardware-software co-design.

4.2 Mapping automotive Use Cases on Optical Computation

The research Section 2.1.1 of this thesis comprehensively examines three pivotal areas, offering a detailed overview of the current advancements in photonic hardware, particularly emphasising its role in facilitating matrix vector multiplication. It highlights the emergence of four promising technologies within the research and startup ecosystem, each distinguished by its unique features and potential applications. The analysis further delves into the advantages and disadvantages associated with each technology. Specifically, it questions the feasibility of using Wavelength Division Multiplexing in automotive settings due to its non-reconfigurable beam steering capabilities, which could lead to diminished precision over time because of continuous vibrations and impacts experienced during vehicle operation. Similarly, the integration of Multiplane Light Conversion into SoC platforms is identified as challenging and cost-intensive, primarily because MPLC has been predominantly validated in free-space contexts with only recent experiments suggesting small matrix-sizes on integrated circuits. The thesis also evaluates the prospects of employing Mach Zehnder Interferometer and crossbar arrays for photonic MVM implementation. While crossbar arrays are recognised for their significant potential in noise and throughput, their scalability to larger matrix sizes and limited applicability in current products are noted as constraints - being at least a topic for further research and engineering effort. Conversely, MZIs are acknowledged for their ongoing integration into startup products and the advantage of economic scalability, attributed to their widespread use in other domains such as network switching [45]. This comprehensive analysis not only sheds light on the technical intricacies and application prospects of these technologies, but also underscores the need for further research to address the identified challenges and optimise their integration and functionality in real-world settings. As of writing this thesis, an application of MZI is considered to be the best performing and best integrate-able solution for a photonic accelerator to use for MVM.

Contrary to a proposed processing unit for improving computational offering, stands the processing demand within an offered vehicle. Section 3 analysed a variety of use cases within the vehicle in regards to certain Service Level Requirement. After identifying possible areas for beneficial use of photonic in Section 3, two selected areas were further analysed for their specific algorithms and architecture in previous chapter 4.1. It depicted that directly apparent wins of applying a photonic accelerator lie within the overall use case of environment perception and rendering virtual content for displays. Those services have greatly distinguished themselves through the need to run locally, to enable a smooth user experience, in case of video rendering or safety manners, in case of perception. Whilst graphical content, in the case of entertainment features is not safety relevant, the correct and reliable detection of other road users is a critical service, categorised as ASIL-D. This means that a photonic accelerator must not "fail" or be unavailable for larger portions of time when in active use. Photonics must further be as noise-free as possible to not negatively impact the accuracy of underlying algorithmic approaches, as seen in analysing the impact of noise on CNN. While the impact of hardware noise on graphical rendering-pipelines is to be fully understood and tested in terms of image quality, first tests with shading yield great hope in possible use of a photonic accelerator. Notably, both identified use cases are highly utilising matrices for processing and representation of data.

Employing a Mach Zehnder Interferometer as an accelerator necessitates a reduction in the numerical bit-precision of data, currently utilised for both rendering and neural network operations. For neural networks, quantization techniques allow for the adaptation to 8-bit precision during inference, following model training with conventional 32-bit precision, as highlighted in the work by NVIDIA (2020) on quantization aware training [174]. This adaptation facilitates the direct implementation of models on photonic MZI hardware accelerators enabling the here discussed precision of 8bit as seen in Section 2.1.1. Similarly, graphic pipelines could potentially reduce bit precision for shading processes to 8 bits, albeit transformations of objects must remain capable of exceeding the 8-bit numerical range of [0-255]. Reducing the precision or depth of lighting may however reduce the perceived quality of a rendered image. This does remain as an open question and is dependent on the specific rendered content.

Another critical factor for consideration is the dimension of processed matrices. Rendering pipelines conventionally utilise matrices of maximum dimensions of 4x4 for transformations and 3x3 for shading. These dimensions are well supported by current photonic hardware implementations, which have demonstrated capabilities for matrix dimensions up to 256x256, as discussed in Section 2.1.1. In contrast, matrix dimensions for fully connected layers within

CNNs and elements of attention-based transformer models, often exceed 1,000, a scale beyond the current hardware scaling capabilities of photonic MVM implementations.

To address this limitation, the concept of matrix partitioning or tiling can be employed. This approach involves decomposing a given matrix-operation into multiple operations of smaller dimensions [114], [161]. This method enables the utilisation of hardware accelerators that support smaller matrix dimensions than those required by the processing tasks within environment perception models.

Integrating quantization and tiling techniques makes it possible to facilitate neural network processing on hardware accelerators with limited hardware capabilities. Although, this may result in a partial sacrifice of the speedup provided by photonic acceleration, the implementation of an MZI based framework becomes a viable solution for processing larger neural networks. This approach underscores the potential for innovative hardware solutions to overcome the constraints of traditional computational models, offering a pathway towards more efficient and scaleable neural network processing. The use and benefit of using noise afflicted hardware for rendering processes must further be explored in experiments. The limited precision (<10 bit) offered by MZI (and other possible photonic approaches) forbids a direct usage for object transformations and limits the use of photonic accelerators to shading processes, where the limited precision might be less noticeable and noise to be less of a problem.

Overall, those considerations and observations lead to a thorough understanding of the requirements for a potential optical accelerator from a service or algorithm point of view. The following key-elements have been identified:

- An accelerator must be able to calculate matrices of different sizes ranging from 3 x 3 to 1,000 x 1,000 and beyond, for usage in larger neural networks. (refer to Section 4.1.1)
- Data-transfer must be fast, as MVM, whilst being a substantial part of algorithms introduced, are used throughout the entire processing chain requiring other (likely digital) calculations throughout.
- Hardware noise should be kept as low as possible to reduce the impact on model accuracy or rendering of images.

An integration into a moving vehicle adds up on those requirements, as hardware within a vehicle is subject to a changing environment and subordinate to security requirements [206]. Electronic components must further be resilient to vibrations and possible shocks, caused by either road or other mechanical components of the vehicle itself (such as an internal combustion engine). As a vehicle is sold across several markets and climate zones and as it is exposed to the nature around it whilst being operated, components must withstand environmental challenges. Whilst assumable, electronic components are shielded from rain or wind, heat and humidity must be withstood by an integrated hardware. A requirement classification of the automotive electronics council (AEC) suggests, that a processing unit enclosed within the vehicle should be able to withstand temperatures of $-40C$ to $+125C$ [21]. MZI and micro-ring-resonator based systems are already proposed with heating-elements within, as minimal noise is achieved above 30 degrees Celsius. Each heater however is adding up to the overall energy consumption of the accelerator (10 mW per heater [302]). This means, that a photonic accelerator within a vehicle must be heated before its use in cold environments or cooled in very hot situations, that can occur when a vehicle is stationary in direct sunlight in hot climate regions [64], [73], [275]. Managing temperature will increase the boot-up delay of a vehicle before safe driving is possible. Embedding heating elements into the photonic accelerator itself, as proposed by Duan et.al. [73], may keep this time within an acceptable magnitude.

The impact of vibration on a photonic accelerator remains an open question. Current hardware implementations are demonstrated within a static context and no known tests have been done, that enable to derive information on this matter. Known however is, that even within a static context, MDM and WDM can suffer from higher noise due to slight imperfections within manufacturing. Their lack of reconfigurability is another indicator, that those technologies might not be perfectly suited for this application. MZI however are known to be quite robust suggesting a good possibility to work in this context [64].

On the positive side of things stands the photonic resilience against electromagnetism. As light does not interfere with magnetism [130], reduced shielding for purely optical elements of the system might be possible. As however only small parts of the full SoC systems are realised photonically, this impact may be little.

A major requirement, especially for the service cluster of autonomous or assisted driving is the fail safeness described within previously mentioned ASIL classifications. Processing power for vehicle operations are classified as ASIL-D [206], requiring a failure rate of less that $1E-10$ /hr. The fail rate of a system is dependent on a great variety of factors as described by Vigrass in [269]. Those include hardware degradation or possible unforeseen environmental impact. A noise afflicted processor must also ensure to deliver a maximum amount of noise within ASIL specifications to ensure, that not only a mathematical operation is executed, but that it is executed with a minimum precision needed for safe operations.

While some of the requirements, such as the resilience to hardware degradation over the vehicles operational time, being difficult to compare with the current phase of first hardware implementations of photonic accelerators, the paper landscape suggests, that a model of MZI is most robust and has the best chances for an application within the automotive sector.

4.3 Basic Architecture of an integrated Optical Accelerator

The emergence of heterogeneous computing and the evolution of SoC technologies described previously in Section 2, enables advanced computational models that integrate diverse processing modalities and capabilities. The proposition to embed a photonic hardware accelerator within a SoC, as illustrated in Figure 17, reflects an approach that seeks to harness the advantages of photonic technology in semiconductor frameworks. This concept draws inspiration from the currently evolving integration of specialised NPU into SoC by corporations such as Apple, Qualcomm, and Tesla [15], [56], [199], [200], [254]. Introducing an OPU is another opportunity to increase processing performance through introduction within an SoC.

The high level architecture outlined in Figure 17 aims to synergize traditional electronic components with a photonic analogue accelerator, incorporating at least a unified memory, a CPU, an input/output controller, and a data transfer method to be specified. This integrated approach is poised to leverage the speed and bandwidth of photonic computations, addressing the growing demands for enhanced computational throughput, energy efficiency, and device miniaturisation.

The extension of a SoC architecture with an OPU represents a strategic fusion of photonic and electronic computing. This Section will further discuss the architectural, functional, and practical aspects of the OPU, contributing to the discourse on advancing computing technologies through photonic integration.

An analysis of the underlying concept of an MZI has already been explored in Section 2.1.2. A single MZI can be used to compute MVM of a 2x2 unitary matrix. The amount of MZIs needed for a unitary input matrix of dimension NxN can be computed through: $N(N-1)/2$ [80]. The algorithmic analysis of potential services to run on an accelerator span greatly from a minimum of 3x3 (shading) to a size greater >1,000 x 1,000. A minimum size for neural networks relies with 64x64 attention mechanisms. Known and already realized matrix accelerators are a potential point of reference for a decision on optimal hardware-enabled matrix sizes. The Google TPU in its first iteration for instance, features a 256x256 matrix-multiply unit (MXU) [218], whilst a more current version is made up of smaller 128x128 units [49].

The dimension of to be processed matrices has a direct influence on both the physical size and energy consumption of the OPU. A single MZI takes up roughly $8,000\mu m^2$ [256] of die size. Considering the 256x256 dimension of input matrices, as promised by Lightmatter [144] and introduced within Google TPU [218], an accelerator would need a total of 32,640 MZI accounting for a die size of $261.12mm^2$. This is without accounting for any signal-converters, additional attenuators, caches et cetera. Figuring a mean power consumption of 10 mW per MZI [256] this accelerator would consume 326.4 W for its photonic part. Assuming its feasibility for production, using newer approaches to MZI implementation, with a consumption of around 2 mW [153], an accelerator could perform at 65.280 W. Reducing the to-be processed matrix size to 128x128 reduces both the energy consumption and area footprint with 8,128 MZI used. This results in an area of $65.024mm^2$ and a 16 W power consumption. A broader overview of matrix dimensions can be found in Table 7 and graphically in appendix 6, which greatly shows the exponential growth of MZI needed for realisation of a linearly increasing input-matrix dimension.

Table 7: Overview of matrix dimensions and energy consumption

Matrix dimension	Amount of MZI	Chip area (mm^2) [1]	MZI Power consumption (W)[2]
256x256	32,640	261.12	65.280
128x128	8,128	65.024	16
64x64	2,016	16.128	4.032
32x32	496	3.968	1.984

[1] Chip size of MZI only without signal conversion and caches

[2] Energy consumed by MZI and respective heaters only with 2 mW assumed per MZI. This metric is to indicate a dimension of power consumption, as the overall SoC power draw is dependent on a great amount of factors like lasers, Analogue Digital Converter (ADC), preprocessing and more.

The basic architecture of a MZI based accelerator is shown in Figure 16. The MZI array itself is defined by two inputs and one output as previously explored in chapter 2.1.2. One being the input vector and the second one being the configuration of each phase shifter within the MZI grid. Data should be pipelined for performance reasons, as reduced data loading time increases the setup time for each clock cycle. The input vector is created through a series of Digital Analogue Converter before encompassed into the MZI-grid. A Digital Analogue Converter (DAC) refers to a light emitter, that takes a digital value as an input and encodes it onto an analogue optical signal.. The analogue output (light) of this is interpreted through an Analogue Digital Converter, before it can further be made use of. A DAC is a photo-detector, that receives an optical, analogue signal and outputs a digital value electronically. Figure 16 shows how three caches are proposed, with each cache serving as a pipeline for either use of the MZI grid or output handling. Notably, the input data must first be converted into a unitary matrix before processing.

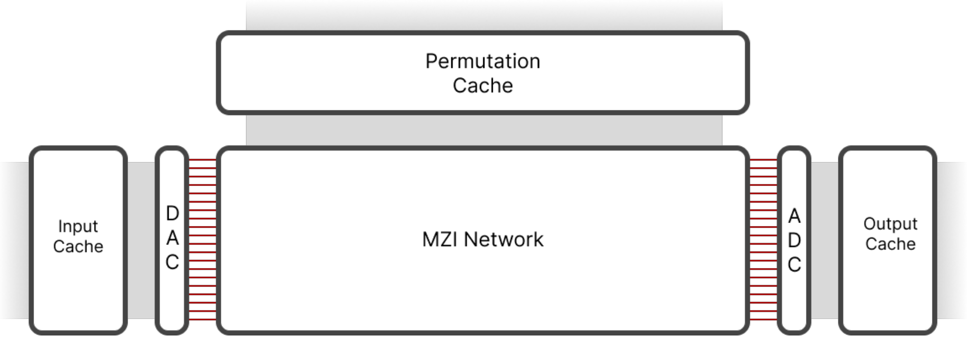


Figure 16: Basic schema of an integrated MZI array
This figure represents the broad concept of a photonic MVM accelerator, that exists of two input caches for matrix and vector, one Digital Analogue Converter, the photonic MZI grid, a photodetector (ADC) and an output cache.

To put required chip size for a certain matrix dimension into perspective, one can compare such with existing accelerators and SoCs. The Tesla FSD 4.0 SoC for instance consumes an area of $260mm^2$ in total, with $12.8mm^2$ being consumed by an ARM Cortex A72 quad-core cluster and the majority of the space being used by two NPUs [280]. A photonic accelerator enabling 256x256 MVM on hardware would consume a die area of $261.12mm^2$ surpassing the entirety of the compared Tesla chip in terms of sizing. This comparison outlines how large optical processing units can become in comparison to electrical hardware.

Comparing an OPU to digital accelerators, such as the Google Tensor Processing Unit (TPU) enables an outlook on the energy efficiency of optical components. The TPU in its fifth generation consumes approximately 40 W for its MXU [49]. Photonic implementations of MVM below 256x256 dimensions could reach significantly lower power consumption, as approximations listed in Table 7 show.

Energy efficiency, while being an important factor, is only comparable, if similar performance is achieved. The performance of the entire photonic system is limited by several items within the processing chain. Whilst the optical components themself can be clocked to speeds up to several THz [159], its input and output greatly limits such capability. A full clock-cycle must (in a simplified overview) offer enough time for:

- Hardware configuration (t_1)
- Input signal stabilisation (t_2)
- "Processing" (t_3)
- analog digital conversion (t_4)

Hardware configuration (t_1) refers to the time needed to configure each phase shifter of the MZI grid to enable the shift needed for desired MVM. Similarly input signal stabilisation (t_2) is the time needed for an input, created by the DAC to become stable and representative of the input needed for ongoing computation. Hardware configuration (t_2) is greatly dependent on the actual material and composition of each phase shifter implemented within the MZI. These two steps may happen in parallel, with each initialisation timing not affecting the other. Processing time (t_3) is described by the time needed from finished setup time (t_1 and t_2) to a stable output signal at the end of the MZI grid. Computation is realised through propagation of light through the grid of MZI at the speed of light ($299792458m/s$), accounting to a time of $34ps$ (assuming a total photonic distance of 1cm, approximated through chip area of a 128x128 dimension). This shows, how the matrix dimension of an accelerator and the associated growth in size already limits the possible maximum clock frequency. With a travel time of the light of $34ps$, a maximum clock cycle of 29.41 GHz is possible with the assumption of fitting an entire computation within a single clock cycle. Lastly, ADC (t_4) refers to the time needed for output conversion.

Research papers for proposed MVM accelerators greatly vary in expected time wise delay for a configuration on a scale from few microseconds [258] to 10 ms [293]. Configuration time however, has a direct impact on the maximum frequency of the overall photonic accelerator, considering a worst case scenario of a changing input matrix for every single clock cycle. Analysing MZIs used for transceivers for optical communication systems reveals faster configuration times, within the lower pico-second ($\sim 10ps$) dimension, thus enabling 100 GHz frequencies [95]. Notably however, picking a photonic phase shifter for MZI design does have an impact on energy consumption and noise. The last elements of the overall structure proposed in Figure 16 is the signal conversion from digital to analogue and vice versa. Similar to the selection of a specific phase shifter, ADC and DAC are selected with a trade-off in energy consumption and speed. Tian et. al. propose a 25 GHz converter [256], whilst Tsirigotis et. al. propose 128 GHz [263](for a smaller 4x4 dimension accelerator).

To enable a 128x128 matrix to vector multiplication an input of 128 lasers (DAC), an output of 128 ADCs are mandatory and three respective caches as mentioned. With a bit-precision of 8 bit per signal, the input and output cache must be at least 1,024 bit large. The third cache feeds the configuration of all embedded phase shifters within the MZI array. This accounting to a size of 130,048 kBit.

Imaging a MZI network for the acceleration of 128x128 matrices, that operates at 28 GHz (limited through size of the MZI grid) enables a performance estimation in Tera operations per second (TOPS) by:

$$TOPS = \frac{f \cdot o \cdot p}{10^{12}}$$

Where:

- f is the operational frequency in Hertz (Hz)
- o represents the number of operations performed per cycle
- p indicates the number of parallel processing units

Utilising this specified formula, the performance of the proposed solution can be calculated to reach 227.58 TOPS without multiplexing and parallelisation. The energy consumption of the chip consists of contributions from both photonic and electronic segments of the accelerator. The photonic components are made up of MZI, input lasers, ADC, and attenuators, cumulatively accounting for an estimated power draw of around 20 watts. This estimate includes a power usage of 16 watts for the MZIs, approximately 768 mW for the ADCs [256], 1152 mW for the lasers [256], and 2 watts for the attenuators embedded within the MZI grid. It is imperative to highlight however, these figures are extrapolated from experimental observations conducted on configurations with smaller matrix dimensions and variable noise levels, rendering the projected energy consumption substantially theoretical. Beyond the optical components, additional power is consumed by electronic elements such as caches, controllers, and any pre-processors, further contributing to the overall energy footprint of the system.

The throughput of the chip is an interesting metric, that is important to keep in mind for further design for integration of such accelerator within an SoC. A single cycle of the photonic hardware consumes a 128x1 vector and a 128x128 matrix with 8 bit precision for each value - 132,096 bits total.

Assuming a clock speed of 28 GHz, this results in a total bitrate of 3,698,688 Gbit/s needed for full utilisation of the proposed hardware accelerator. The bitrate for data transfer could potentially be reduced through previous encoding of data and decoding within the accelerator. The high clockrate of the processor could further be used for running the same computation multiple times, thus reducing the impact of hardware noise through building a mean average result of >1 computation. With a theoretical datarate of >3 Terabit/s for processing it is questionable, if this high clock rate can be utilised at all. The question of possible memory connections will be answered within Section 4.4 with a focus on possible memory integration using examples from other hardware accelerators.

Memory throughput and data transfer rates present a crucial parameter to design and integration of a possible optical accelerator. Within a singular operational cycle (assuming no parallelism), the photonic hardware is used to process a vector of dimensions 128x1 and a matrix of 128x128, each value represented with 8 bit precision. This culminates in a total data volume of 132,096 bits. Predicated on an operational frequency of 28 GHz, this computational demand translates into a requisite data throughput of 3,698,688 Gbit/s to achieve full utilisation of the proposed hardware accelerator's capabilities. It is conceivable that the requisite bitrate for data transmission could be effectively mitigated, through the implementation of advanced data encoding techniques prior to input and subsequent decoding processes within the accelerator framework, thereby enhancing data handling efficiency and effective data rates [291].

4.4 Integration of photonic Accelerator

An open question after proposing a general architecture of the photonic accelerator itself, remains within its integration. Due to the fact, that a grid of MZI are only able to process unitary matrix-vector computation, several pre-processing steps are needed to enable the use of such. According to the concept of singular value decomposition (SVD), a matrix A can be decomposed as $A = U\Sigma V^*$ [244]. This decomposition is found within previously shown MZI grid in Section 2.1.2. If the objective is to enable Matrix Matrix multiplication on the hardware accelerator, a decomposition into several vector matrix multiplications is needed. As the decomposition of a matrix into a unitary representation for usage within a photonic MZI accelerator is needed for every single computation, it makes sense to add this step to the accelerator hardware itself. This could be realised through the implementation with specialised hardware such as a Field-programmable Gate Array (FPGA) setup for instance, as also proposed by Agrafiotis et. al. [7].

With the data-rates needed for full utilisation of the proposed optical accelerator, a major question for integration relies on the memory integration. Currently known hardware accelerators in the optical and digital domain serve as examples for technical possibilities.

Within the optical domain, both startup Lightmatter and Lightelligence offer an extension card for matrix operations, that is connected through PCIe [142], [279]. In its latest version (6.0), PCIe promises a maximum throughput of 64 GigaTransfers/s and up to 256 GB/s through 16 lanes [188]. Devices supporting this standard however have yet to appear on the market, but are expected to arrive within the year of 2024 [53]. Lightmatter is known to be using PCIe 4.0 enabling better compatibility with state-of-the-art hardware available on the market [67], which limits transfer speeds to 32 GB/s [127] and therefore negatively impacts the theoretical speed of the photonic accelerator. Similarly the Coral AI SoC offered by Google uses PCIe 2.0 for integration of the TPU-Edge further limiting maximum transfer speeds [54].

An alternative seen within the FSD-Chip built into Tesla vehicles, that uses 2 NPUs for acceleration is the data transfer through a Network on Chip (NoC) [281], as depicted in Figure 17. For complex SoCs, a NoC can provide a scaleable and efficient solution for data transfer between storage and processing units, as they are able to use a shared hardware architecture for routing data. The on chip integration of a NoC, with direct memory access of multiple cores, is proven to provide greater speeds through higher data throughput and a reduced number of memory accesses in comparison to PCIe solutions [40], [257], [261]. Known implementations of NoC deliver connection speeds of several GB/s. AMD manuals [13] propose throughput of 34 GB/s per memory controller, enabling horizontal scaling effects when sacrificing chip-area. Park et. al. promise a theoretically achievable data rate of 892 Gb/s (111,5 GB/s) [184]. The beneficial properties of photonics are also being introduced to NoCs to further increase throughput and reduce energy consumption [24], [189]. A photonic realisation enables beyond terabit datarates with a significantly reduced energy consumption. Paper [24] proposes a crossbar array, as already introduced within Section 2.1.2 with a bandwidth of 1.92 Tbit/s (240 GB/s). The concept of photonic Network on Chip is already proven within the Lightelligence optical accelerator that, whilst relying on PCIe for Server integration, uses a photonic NoC for on-chip communication. Another option for memory integration is the usage of High Bandwidth Memory (HBM), that enables direct integration of an accelerating processing unit with high performance memory. Chip manufacturer Micron has announced production of a HBM iteration 3 memory, that enables throughput of >1.2 TB/s (9600 GBit/s) with plans to increase bandwidth beyond 2 TB/s in upcoming years [230] and marks best ranking performance. HBM is already used for AI accelerators such as the Google TPU [119].

Figure 17 shows a block-diagram of a high-level architecture of an SoC, that uses a NoC for combining multiple parts of different modality within the chip. Those include a (or many) CPU core, a (or many) GPU core, an OPU, an input/output controller and unified memory. The exact amount of each core and the sizing of used memory remains open for the possible adaption to further Hardware / Software codesign.

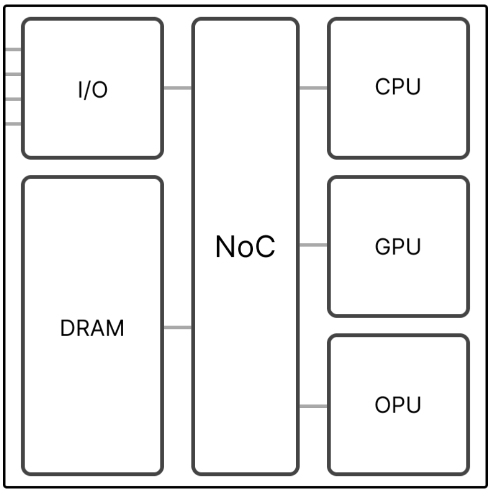


Figure 17: High-level overview of OPU chiplet integration
As per example by current SoC implementations, one could eventually integrate an OPU into a single die. This image represents a possible high-level block diagram for a possible SoC, that could be used for environment perception. It features a unified memory, an input / output controller, a CPU, GPU, NoC and proposed OPU

Further options for connecting a hardware accelerator to memory are to utilise a BUS-system, like ARM's Advanced Microcontroller Bus Architecture (AMBA), direct memory access (DMA), or serial interfaces (Serial Peripheral Interface (SPI), Inter-Integrated Circuit (I2C)). Realisations of a bus system for ARM systems have been around for several years with the AMBA protocol family first being released in 1996 [151] and being updated in several revisions since then. Current implementations of such a bus-system are proposed with throughput of up to 6,152 Mb/s (0.769 GB/s) [214], [273]. Serial interfaces are enabling a possible throughput significantly below those of bus-systems, and therefore do not make any sense for consideration [225]. Another option to connect proposed hardware accelerator with memory, is the usage of DMA. This enables a direct connection of the OPU with the memory offering data transfer with speeds up to 248 Gbit/s (31 GB/s) [101]. Currently available ARM NPUs are equipped with DMA for memory access [18], whilst upcoming designs are proposed with NoC data transfers [106]. The Tesla FSD 4.0 chip once again marks a great example, as it shows how within the automotive sector an SoC is already integrated with a NPU interconnected with a NoC.

With the data throughput needed to feed a previously theoretical OPU exceeding possible maximum throughput of known hardware possibilities to connect it to memory, there are two possibilities to overcome such issues. Parallelisation of memory connections is one way to enhance the overall throughput. This could be realised by combining multiple DMA connections to different caches within the OPU, that do however make the architecture of such more complex. Another option is to decrease the clock rate of the hardware accelerator reducing the theoretical throughput and enable application of known memory connectors. This approach is supported by the fact, that both Lightelligence and Lighmatter are employing greatly lower system clock speeds, of sub 2 GHz [64], [143]. This is also reasoned with complications of ADC at the time of paper release in 2021, when referenced technology would fail to deliver accurate results above 2 GHz [64], even though used components would theoretically be able to perform at up to 10 GHz. Table 8 summarises the effect of a decrease of clock speeds on TOPS (performance), memory throughput and efficiency, which is graphically represented in Appendix 6 Notably however, here depicted throughput represents a worst-case scenario of ever changing input data for each clock cycle.

Table 8: Computed TOPS and Memory Throughput for various Clock Frequencies

Clock-Frequency (GHz)	TOPS	Memory Throughput (Gb/s)	TOPS/W
28	227.584	3,698,688	14.22
20	162.56	2,641,920	10.16
10	81.28	1,320,960	5.08
5	40.64	660,480	2.54
2	16.256	264,192	1.02
1	8.128	132,096	0.51
0.1	0.812	13,209.6	0.051
0.05	0.406	6,604	0.0254
0.01	0.081	1,320.96	0.005

Comparing the worst case required memory throughput of the OPU with previously identified maximum data transfer rates of different memory connectors shows, how the reduction of the accelerators clock cycle enables a feasible integration onto a SoC. A frequency of 10 MHz reduces throughput to a dimension serve able with a direct HBM integration. Even advanced photonic NoCs would only allow for clock rates of <10 MHz due to bandwidth limitations. Noteworthy, using decreased OPU system clock speeds (in terms of data loading) enables to make use of the concept to compute the same numbers more than once and calculate the mean average above those, thus reducing the noise of the MZI grid. The photonic detection rate is however still limited to the maximum frequency of integrate able ADCs.

To assess the efficiency and performance of computational units, a comparison between the proposed OPU and Google's TPU version 4, which was introduced last year, is insightful [49], [264]. The Google TPUv4 boasts a computational capacity of 275 TeraFLOPS (TFLOPS) at a power consumption rate of 170 watts, operating at a frequency of 1050 MHz. This configuration yields an efficiency of 1.62 FLOPS per watt. Additionally, each TPU is equipped with high-bandwidth memory, capable of achieving a data transfer rate of up to 9600 Gb/s [119]. In contrast, a variant of the TPU designed for edge computing applications enhances efficiency to 2 TOPS per watt, albeit at a reduced overall performance of 4 TOPS [113].

Table 8 presents information on the performance capabilities of an OPU, specifically focusing on MVM operations. Notably, the OPU achieves approximately 0.8 TOPS at a clock speed of 0.1 GHz, with a power consumption of 16 watts, resulting in an efficiency of 0.5 TOPS per watt. This marks a significant decrease in efficiency compared to the TPU. However, the direct comparison of TOPS as a metric for performance is contentious, given that it requires a uniform definition of an operation. TPUs utilise a systolic array architecture for MVM operations, which necessitates twice the number of operations for a single MVM compared to the MZI array used in OPUs [218]. Notably, the energy consumption of an OPU is greatly static due to used heaters that are used independently from the system clock speed. Taking this into consideration, OPUs may exhibit superior performance relative to TPUs, but lack an energy-efficiency with current limitations. Furthermore, the TPU's design is not limited to MVM tasks alone. It incorporates additional on-chip logic, further augmenting its operational capabilities.

A notable difference in the comparison held before is the high delta between the maximum possible data transfer rate of a Google TPU and (worst case) assumptions on data rates needed for an optical accelerator. As mentioned within analysing the impact of clock rates on performance, the data rate is a worst case extrapolation for the case, when each clock cycle would require new data to process. Realistically weight matrices (NN) or transformation matrices (for graphics) are used more than once and therefore significantly reduce the needed data transfer for each clock cycle of the OPU.

5 Conclusion

The objective of this thesis has been to grasp an understanding of the underlying physics of photonic computation for an eventual usage as an OPU and to analyse its potential in increasing computational performance for vehicle use cases.

The research Section has highlighted the ongoing interest into optical matrix vector multiplication (MVM) and analysed four approaches to enabling this within hardware. Mach Zehnder Interferometer are a concept already proven for an integration within SoCs that, whilst mostly still being in an early research phase within startups, beckons great aspiration for fulfilling both automotive and usability requirements, that include sufficient bitwise precision, matrix dimensions and resilience to hardware noise. A photonic accelerator promises great performance and throughput with a relatively low energy consumption, as presented within this thesis. An OPU however is used best to its potential at higher clock rates. The introduction of multiplexing is another major chance for increasing performance of up to 500x over compared Google TPUs [182].

An investigation of vehicle services, relying on computational performance for either safety, vehicle autonomy or customer use cases yields two services, that seem like a great chance for acceleration, due to their high data throughput and processing demand. The perception of a vehicle environment that is based on the processing of several sensors, such as camera or LIDAR, input is one of these use cases. Promising approaches to do so are the use of NN-architectures or transformer models. Both heavily utilise MVM operations. Rendering graphical content for in-vehicle displays marks the second use case identified, due to the algorithmic complexity and pixel dimensions of content. Once again, underlying algorithms heavily employ MVM. An open question to application remains with the resilience of an algorithm to a loss of precision and the introduction of seemingly random noise. Simulations for the use within rendering processes have shown limited use for transformations, but promising results for shader processes, requiring further research into possible post processing. The research landscape greatly describes the impact on the use for NN consisting of a relatively small decrease in model accuracy. Once again asking for enhanced model specific experiments.

With great hope and possibility for MVM based acceleration through optical hardware accelerators, the thesis delivered a deeper insight into ongoing trends of heterogeneous computation and SoC integration, highlighting the benefits in efficiency and performance. Theoretical performance of an OPU is limited by various factors, that include the setup time of the MZI grid, detection speed of signals (the ADC) and data transfer speeds ,ultimately limiting the amount of data to be processed by an integrated hardware accelerator. A summary of technological possibilities highlights the advancement of applying a NoC for integration of an OPU due to its direct connection with other processing units and transfer rates. Notably, even data rates promised within currently ongoing research for application of photonic NoC concepts, would limit an OPU to a clock-rate below its theoretical capabilities. An Optical Processing Unit (OPU) is achievable and can be integrated with HBM for data transfer. This feasibility is supported by comparisons with already integrated hardware, and the OPU could operate at a clock speed of less than 0.1 GHz.. This would result in a performance of <1 TOPS (MZI operations). The critical point to consider however is, that this low performance is approximated for a worst case scenario in terms of changing data per operation.

Lastly, the possible utilisation of an OPU is dependent on the hardware-software codesign that is meant to best make use of hardware capabilities. This includes considerate pipelining and reduction of setup times through aggregation of same matrix to changing vectors MVM, as it is the case with using the same transformation matrix for changing objects within rendering pipelines.

Overall: The benefit of integrating an OPU is questionable with electonic hardware accelerators achieving great efficiency and performance. The inherent hardware noise associated with analogue, photonic signals poses challenges for transformation processes within rendering pipelines. However, this noise does not significantly impact AI use cases or shaders, where the unique attributes of OPUs could offer advantages.

Despite these potential benefits, photonics technology remains largely in the experimental phase. Moreover, with the current limitations in memory bandwidth, integrating OPUs as a superior alternative to existing NPUs presents a formidable challenge. The bottleneck created by memory bandwidth constraints complicates the effective utilisation of OPUs, making their comparative advantage over traditional NPUs less clear in current applications.

6 Outlook

This thesis has greatly summarized the possible advancements of integrateable hardware accelerators, that could become part of state-of-the-art vehicle hardware, as represented with the integration of NPUs within Tesla vehicles and Mercedes concepts. It is also a good example for an ongoing trend towards custom chip design for application specific domains attributing the prediction of Gartner, that a large amount of best performing OEMs would delve into such endeavour by 2025 [92]. Increasing possibilities for SoC integration is likely to further push this trend through opportunity of multi-die size and three-dimensional packaging. As of now, photonic accelerators and its underlying components remain within a low technology readiness level. Ongoing experimental tape-outs and first products launching, are a great sign for enabling and improving optical technology while data transfer and optimized usage of hardware remain a significant bottleneck for its effective use. Selection of a specific software architecture to tackle environment perception will enable more considerate cache design and pipe-lining processes, as a transformer based approach yields different optimization than a deep NN for an end to end training concept. Questionable remains, if a photonic accelerator is able to outperform electric ones like the Google TPU improving in efficiency and performance within every release. Potentials do however lie in ideas such increasing memory transfer rates with improvements in HBM. Possibly, optical memory or optical data transfer within systems [12], [128], [272] may eliminate the need for ADCs and enable more performance and greater efficiency, whilst increasing the amount of purely photonic components of the processing chain.

References

[1] 3E8. *HOME | 3E8*. https://www.3e8.co/. Accessed: 08.02.2024.

[2] 5G Automotive Association. *C-V2X Use Cases and Service Level Requirements - Volume I*. https://5gaa.org/c-v2x-use-cases-and-service-level-requirements-volume-i/. Accessed: 30.11.2023. 2020.

[3] 5G Automotive Association. *C-V2X Use Cases and Service Level Requirements - Volume III*. https://5gaa.org/c-v2x-use-cases-volume-ii-examples-and-service-level-requirements//. Accessed: 30.11.2023. 2021.

[4] 5G Automotive Association. *C-V2X Use Cases and Service Level Requirements - Volume III*. https://5gaa.org/c-v2x-use-cases-and-service-level-requirements-volume-iii//. Accessed: 30.11.2023. 2022.

[5] MB ABSTS. *Airbag Sensor Test System*. Accessed: 01.02.2024. URL: https://www.mbdynamics.com/wp-content/uploads/2016/03/ABSTS-Data-Sheet-2009.01.14.pdf.

[6] Michael Aeberhard and Torsten Bertram. "Object Classification in a High-Level Sensor Data Fusion Architecture for Advanced Driver Assistance Systems". In: *2015 IEEE 18th International Conference on Intelligent Transportation Systems*. 2015, pp. 416–422. DOI: 10.1109/ITSC.2015.76.

[7] Georgios Agrafiotis, Eftychia Makri, Ilias Kalamaras, et al. "Nearest Unitary and Toeplitz matrix techniques for adaptation of Deep Learning models in photonic FPGA". In: *Proceedings of the Northern Lights Deep Learning Workshop* 4 (Jan. 2023). DOI: 10.7557/18.6825.

[8] Mohamed Ahmed, Alexandr Klimchik, and Riby Boby. "Fusion of Data from Lidar and Camera in Self Driving Cars". In: Aug. 2022.

[9] Celestial AI. *Celestial AI - Homepage*. https://www.celestial.ai/. Accessed: 08.02.2024.

[10] AkheTonics. *THE FIRST ALL-OPTICAL XPU*. Accessed: 12.12.2023. 2023. URL: https://www.akhetonics.com/.

[11] Paul Alcorn. *Intel's CEO Says Moore's Law Is Slowing to a Three-Year Cadence, But It's Not Dead Yet*. Tom's Hardware. Accessed: 05.01.2024. 2023. URL: https://www.tomshardware.com/tech-industry/semiconductors/intels-ceo-says-moores-law-is-slowing-to-a-three-year-cadence-but-its-not-dead-yet.

[12] Theoni Alexoudi, George Theodore Kanellos, and Nikos Pleros. "Optical RAM and integrated optical memories: a survey". In: *Light: Science & Applications* 9.1 (May 2020). ISSN: 2047-7538. DOI: 10.1038/s41377-020-0325-9. URL: http://dx.doi.org/10.1038/s41377-020-0325-9.

[13] AMD. *XILINX - Memory and Data Movement*. Version 2022.1. Accessed: 2024-02-05. 2022. URL: https://docs.xilinx.com/r/2022.1-English/ug1504-acap-system-solution-planning-methodology/Memory-and-Data-Movement.

[14] European Space Agency Antonio Franchi. "Impulse Statement by the European Space Agency". European Space-enabled Connectivity Solutions for the Car of the Future. 2023. URL: https://www.german-ba-ambassador.de/s-projects-side-by-side.

[15] Apple Inc. *Apple at Work - M1 Overview*. https://www.apple.com/ua/business/mac/pdf/Apple-at-Work-M1-Overview.pdf. Accessed: 30.01.2024. 2024.

[16] ARM. *Affordability for Custom SoC*. https://armkeil.blob.core.windows.net/developer/Files/pdf/white-paper/affordability-for-custom-soc.pdf. Accessed: 01.02.2024. 2020.

[17] ARM. "ARM Expects Vehicle Compute Performance to Increase 100x in Next Decade". In: *ARM* (2023). Accessed: 30.01.2024. URL: https://www.arm.com/company/news/2015/04/arm-expects-vehicle-compute-performance-to-increase-100x-in-next-decade.

[18] Arm Limited. *Functional Description: Functional Blocks*. Accessed: 2024-02-05. Arm Limited. 2020. URL: https://developer.arm.com/documentation/102420/0200/Functional-description/Functional-blocks-.

[19] TechWire Asia. *AMD Revolutionizes Auto Tech: Unveiling Next-Gen AI Engines at CES 2024*. https://techwireasia.com/01/2024/proofed-amd-revolutionizes-auto-tech-unveiling-next-gen-ai-engines-at-ces-2024/. Accessed: 30.01.2024. 2024.

[20] Audi AG. *A showcase becomes a reality: Audi brings VR experience platform to CES 2023*. Audi MediaCenter Press Release. Accessed: 12.12.2023. Jan. 2023. URL: https://www.audi-mediacenter.com/en/press-releases/a-showcase-becomes-a-reality-audi-brings-vr-experience-platform-to-ces-2023-15106.

[21] *Automotive Electronics Council (AEC) - Q200 Rev D: Stress Test Qualification for Passive Components*. Tech. rep. Accessed: 2024-02-01. Automotive Electronics Council, 2010. URL: http://www.aecouncil.com/Documents/AEC_Q200_Rev_D_Base_Document.pdf.

[22] Rodrigo Ayala and Tauheed Khan Mohd. "Sensors in Autonomous Vehicles: A Survey". In: *Journal of Autonomous Vehicles and Systems* 1.3 (Dec. 2021), p. 031003. ISSN: 2690-702X. DOI: 10.1115/1.4052991. eprint: https://asmedigitalcollection.asme.org/autonomousvehicles/article-pdf/1/3/031003/6818010/javs_1_3_031003.pdf. URL: https://doi.org/10.1115/1.4052991.

[23] Irfan Baftiu, Arbnor Pajaziti, and Ka C. Cheok. "Multi-mode surround view for ADAS vehicles". In: *2016 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS)*. 2016, pp. 190–193. DOI: 10.1109/IRIS.2016.8066089.

[24] Moez Balti and Abderrazak JEMAI. "Performance survey of classic and Optic network-on-chip". In: *IET Circuits, Devices & Systems* 15.4 (Mar. 2021), 393–402. ISSN: 1751-8598. DOI: 10.1049/cds2.12025. URL: http://dx.doi.org/10.1049/cds2.12025.

[25] Emily Blem, Jaikrishnan Menon, Thiruvengadam Vijayaraghavan, et al. "ISA Wars: Understanding the Relevance of ISA Being RISC or CISC to Performance, Power, and Energy on Modern Architectures". In: *ACM Trans. Comput. Syst.* 33.1 (Mar. 2015). ISSN: 0734-2071. DOI: 10.1145/2699682. URL: https://doi.org/10.1145/2699682.

[26] Caroline Blight. *Salience Labs advances its AI agenda using new chip design*. https://www.theceomagazine.com/business/innovation-technology/ai-salience-chip-design/. Accessed: 08.02.2024. Apr. 2023.

[27] BMW. *BMW Autonomous Driving*. Accessed: 04.12.2023. 2023. URL: https://www.bmw.com/en/automotive-life/autonomous-driving.html.

[28] O. Eckart BMW AG / 5G Automotive Association. "No connection is not an option." European Space-enabled Connectivity Solutions for the Car of the Future. 2023. URL: https://www.german-ba-ambassador.de/s-projects-side-by-side.

[29] N. De Mattia BMW Blog. *BMW Is Looking Into Satellite-Based Internet Connections*. https://www.bmwblog.com/2022/12/09/bwm-is-looking-into-satellite-based-internet-connections/. Accessed: 08.12.2023. 2022.

[30] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. "YOLOv4: Optimal Speed and Accuracy of Object Detection". In: *CoRR* abs/2004.10934 (2020). arXiv: 2004.10934. URL: https://arxiv.org/abs/2004.10934.

[31] Patrick Bowen, Guy Regev, Nir Regev, et al. "Analog, In-memory Compute Architectures for Artificial Intelligence". In: *arXiv preprint arXiv:2302.06417* (2023).

[32] Daniel Braun. *WebGL-Editor*. https://web.cs.upb.de/cgvb/WebGLEditor/. Accessed: 03.01.2024. 2017.

[33] Daniel Brunner and Demetri Psaltis. "Competitive photonic neural networks". In: *Nature Photonics* 15 (May 2021), pp. 323–324. DOI: 10.1038/s41566-021-00803-0.

[34] John Burkardt. *List of open source object files*. https://people.sc.fsu.edu/~jburkardt/data/obj/. Accessed: 03.01.2024. 2022.

[35] Jeffrey Burt. *Luminous Shines A Light On Optical Architecture For Future AI Supercomputer*. https://www.nextplatform.com/2022/03/17/luminous-shines-a-light-on-optical-architecture-for-future-ai-supercomputer/. Accessed: 08.02.2024. Mar. 2022.

[36] Samuel R. Buss. *3D Computer Graphics A Mathematical Introduction with OpenGL*. May 2019.

[37] Cem. “Automotive Digital Transformation in ’23: Trends & Use Cases”. In: *AIMultiple* (2023). Accessed: 20.12.2023. URL: `https://research.aimultiple.com/digital-transformation-automotive/`.

[38] Cerebras. *Cerebras - Homepage*. `https://www.cerebras.net/`. Accessed: 08.02.2024.

[39] Cheng Chang, Jiawei Zhang, Kunpeng Zhang, et al. “BEV-V2X: Cooperative Birds-Eye-View Fusion and Grid Occupancy Prediction via V2X-Based Data Sharing”. In: *IEEE Transactions on Intelligent Vehicles* 8.11 (2023), pp. 4498–4514. DOI: `10.1109/TIV.2023.3293954`.

[40] Kun-Chih (Jimmy) Chen, Masoumeh Ebrahimi, Ting-Yi Wang, et al. “NoC-based DNN accelerator: a future design paradigm”. In: *Proceedings of the 13th IEEE/ACM International Symposium on Networks-on-Chip*. NOCS ’19. Association for Computing Machinery, 2019. ISBN: 9781450367004. DOI: `10.1145/3313231.3352376`. URL: `https://doi.org/10.1145/3313231.3352376`.

[41] Ming-Fa Chen, Fang-Cheng Chen, Wen-Chih Chiou, et al. “System on Integrated Chips (SoIC(TM) for 3D Heterogeneous Integration”. In: *2019 IEEE 69th Electronic Components and Technology Conference (ECTC)*. 2019, pp. 594–599. DOI: `10.1109/ECTC.2019.00095`.

[42] Junwei Cheng, Hailong Zhou, and Jianji Dong. “Photonic Matrix Computing: From Fundamentals to Applications”. In: *Nanomaterials* 11.7 (June 2021), p. 1683. ISSN: 2079-4991. DOI: `10.3390/nano11071683`. URL: `http://dx.doi.org/10.3390/nano11071683`.

[43] Qixiang Cheng, Meisam Bahadori, Madeleine Glick, et al. “Recent advances in optical technologies for data centers: a review”. In: *Optica* 5.11 (2018), pp. 1354–1370.

[44] Qixiang Cheng, Meisam Bahadori, Madeleine Glick, et al. “Recent advances in optical technologies for data centers: a review”. In: *Optica* 5.11 (Nov. 2018), pp. 1354–1370. DOI: `10.1364/OPTICA.5.001354`. URL: `https://opg.optica.org/optica/abstract.cfm?URI=optica-5-11-1354`.

[45] Qixiang Cheng, Sébastien Rumley, Meisam Bahadori, et al. “Photonic switching in high performance datacenters”. In: *Optics express* 26.12 (2018), pp. 16022–16043.

[46] Max A. Cherney. *Why YouTube decided to make its own video chip*. Accessed: 04.01.2024. Aug. 2022. URL: `https://www.protocol.com/enterprise/youtube-custom-chips-argos-asics`.

[47] Mark Cheung, John Shi, Oren Wright, et al. “Graph Signal Processing and Deep Learning: Convolution, Pooling, and Topology”. In: (July 2020).

[48] Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, et al. *TransFuser: Imitation with Transformer-Based Sensor Fusion for Autonomous Driving*. 2022. arXiv: 2205.15997 [cs.CV].

[49] Google Cloud. *System Architecture: TPU VM*. https://cloud.google.com/tpu/docs/system-architecture-tpu-vm. Accessed: 03.02.2024.

[50] CNBC. *Intel says Moore's Law is still alive, Nvidia says it's ended*. CNBC. Accessed: 05.01.2024. 2022. URL: https://www.cnbc.com/2022/09/27/intel-says-moores-law-is-still-alive-nvidia-says-its-ended.html.

[51] Cognifiber. *Technology - Cognifiber - Homepage*. https://www.cognifiber.com/technology/. Accessed: 08.02.2024.

[52] McKinsey & Company. "Hands off: Consumer perceptions of advanced driver assistance systems". In: *McKinsey & Company* (2023). Accessed: 20.12.2023. URL: https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/hands-off-consumer-perceptions-of-advanced-driver-assistance-systems.

[53] Matthew Connatser. "PCIe 6 to Launch in 2024, PCIe 7 in 2027". In: (2023). Accessed: 05.02.2024. URL: https://www.xda-developers.com/pcie-6-to-launch-in-2024-pcie-7-in-2027/.

[54] *Coral System-on-Module (SoM) Datasheet*. https://coral.ai/docs/som/datasheet/. Accessed: 05.02.2024. 2023.

[55] Sony Corporation. *Sony IMX490 Image Sensor Datasheet*. https://www.sony-semicon.com/files/62/pdf/p-15_IMX490.pdf. Accessed: 04.12.2023. 2019.

[56] N. Cristovao. *Tesla FSD Hardware 4.0 Revealed: More Cameras, New Placements*. Accessed: 04.12.2023. 2023. URL: https://www.notateslaapp.com/software-updates/upcoming-features/id/1205/tesla-fsd-hardware-4-0-revealed-more-cameras-new-placements.

[57] N. Cristovao. *Tesla's FSD hardware 4.0 to use cameras with LED flicker mitigation*. Accessed: 04.12.2023. 2022. URL: https://www.notateslaapp.com/news/679/tesla-s-fsd-hardware-4-0-to-use-new-cameras.

[58] I. Cutress. "Samsung Foundry: 2nm Silicon in 2025". In: (2023). Accessed: 06.01.2024. URL: https://www.anandtech.com/print/16995/samsung-foundry-2nm-silicon-in-2025.

[59] Ian Cutress. *Intel Hybrid CPU "Lakefield": All You Need to Know*. https://www.anandtech.com/show/15877/intel-hybrid-cpu-lakefield-all-you-need-to-know. Accessed: 29.01.2024. 2020.

[60] DataCamp. *An Introduction to Convolutional Neural Networks (CNNs)*. DataCamp Tutorial. Accessed: 20.01.2024. 2023. URL: https://www.datacamp.com/tutorial/introduction-to-convolutional-neural-networks-cnns.

[61] Mike Davies, Andreas Wild, Garrick Orchard, et al. “Advancing neuromorphic computing with loihi: A survey of results and outlook”. In: *Proceedings of the IEEE* 109.5 (2021), pp. 911–934.

[62] Joey De Vries. “Learn opengl”. In: *Licensed under CC BY* 4 (2015).

[63] Cansu Demirkiran, Furkan Eris, Gongyu Wang, et al. “An Electro-Photonic System for Accelerating Deep Neural Networks”. In: *CoRR* abs/2109.01126 (2021). arXiv: `2109.01126`. URL: `https://arxiv.org/abs/2109.01126`.

[64] Cansu Demirkiran, Furkan Eris, Gongyu Wang, et al. “An electro-photonic system for accelerating deep neural networks”. In: *ACM Journal on Emerging Technologies in Computing Systems* 19.4 (2023), pp. 1–31.

[65] Jiajun Deng, Shaoshuai Shi, Peiwei Li, et al. *Voxel R-CNN: Towards High Performance Voxel-based 3D Object Detection*. 2021. arXiv: `2012.15712 [cs.CV]`.

[66] Li Deng. “The mnist database of handwritten digit images for machine learning research”. In: *IEEE Signal Processing Magazine* 29.6 (2012), pp. 141–142.

[67] Maria Deutscher. “Light-based AI chip startup Lightmatter raises $80M in round backed by GV”. In: (May 2021). Accessed: 05.02.2024. URL: `https://siliconangle.com/2021/05/06/light-based-ai-chip-startup-lightmatter-raises-80m-round-backed-gv/`.

[68] Johann A. Dirdal. “End-to-end learning and sensor fusion with deep convolutional networks for steering an off-road unmanned ground vehicle”. In: 2018. URL: `https://api.semanticscholar.org/CorpusID:70094434`.

[69] Bowei Dong, Samarth Aggarwal, Wen Zhou, et al. “Higher-dimensional processing using a photonic tensor core with continuous-time data”. In: *Nature Photonics* 17.12 (Oct. 2023), 1080–1088. ISSN: 1749-4893. DOI: `10.1038/s41566-023-01313-x`. URL: `http://dx.doi.org/10.1038/s41566-023-01313-x`.

[70] Zichao Dong, Hang Ji, Xufeng Huang, et al. “PeP: a Point enhanced Painting method for unified point cloud tasks”. In: *arXiv preprint arXiv:2310.07591* (2023).

[71] Jason Dorrier. “Moore’s Law: Scientists Just Made a Graphene Transistor Gate the Width of an Atom”. In: *Singularity Hub* (Mar. 2022). URL: `https://singularityhub.com/2022/03/13/moores-law-scientists-just-made-a-graphene-transistor-gate-the-width-of-an-atom/`.

[72] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, et al. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. 2021. arXiv: `2010.11929 [cs.CV]`.

[73] Fei Duan, Kai Chen, Da Chen, et al. “Low-power and high-speed 2×2 thermo-optic MMI-MZI switch with suspended phase arms and heater-on-slab structure”. In: *Optics Letters* 46 (Dec. 2020). DOI: `10.1364/OL.413747`.

[74] Europe ESim. *How much data does Google Maps use? | eSIM Europe — europeesim.com*. https://europeesim.com/blog/how-much-data-does-google-maps-use/. Accessed 08.02.2024. Mar. 2023.

[75] United Nations Economic Commission for Europe. *Proposal for amendments to UN Regulation No. 79*. Report GRVA-05-07r3e. United Nations Economic Commission for Europe, 2020. URL: https://unece.org/DAM/trans/doc/2020/wp29grva/GRVA-05-07r3e.pdf.

[76] Z Ezziane. "DNA computing: applications and challenges". In: *Nanotechnology* 17.2 (2005), R27.

[77] Federico Faggin. "How we made the microprocessor". In: *Nature Electronics* 1.1 (2018), pp. 88–88.

[78] Jichao Fan, Yingheng Tang, and Weilu Gao. "Universal Approach for Calibrating Large-Scale Electronic and Photonic Crossbar Arrays". In: *Advanced Intelligent Systems* 5.10 (2023), p. 2300147.

[79] Johannes Feldmann, Nathan Youngblood, Maxim Karpov, et al. "Parallel convolutional processing using an integrated photonic tensor core". In: *Nature* 589.7840 (2021), pp. 52–58.

[80] S. A. Fldzhyan, M. Yu. Saygin, and S. P. Kulik. "Optimal design of error-tolerant reprogrammable multiport interferometers". In: *Opt. Lett.* 45.9 (May 2020), pp. 2632–2635. DOI: 10.1364/OL.385433. URL: https://opg.optica.org/ol/abstract.cfm?URI=ol-45-9-2632.

[81] Nicolas Fontaine, Joel Carpenter, Simon Gross, et al. "Photonic Lanterns, 3-D Waveguides, Multiplane Light Conversion, and Other Components That Enable Space-Division Multiplexing". In: *Proceedings of the IEEE* PP (Nov. 2022), pp. 1–14. DOI: 10.1109/JPROC.2022.3207046.

[82] Martin Forsythe. "Matrix processing with nanophotonics". In: *Medium* (2019). Accessed: 25.12.2023. URL: https://medium.com/lightmatter/matrix-processing-with-nanophotonics-998e294dabc1.

[83] Adi Fuchs and David Wentzlaff. "The accelerator wall: Limits of chip specialization". In: *2019 IEEE International Symposium on High Performance Computer Architecture (HPCA)*. IEEE. 2019, pp. 1–14.

[84] Rohith Gandhi. *R-CNN, Fast R-CNN, Faster R-CNN, YOLO – Object Detection Algorithms*. https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e. Accessed: 22.01.2024. 2018.

[85] Qing Hong Gao, Tao Ruan Wan, Wen Tang, et al. *An Improved Augmented Reality Registration Method Based on Visual SLAM*. Ed. by Feng Tian, Christos Gatzidis, Abdennour El Rhalibi, et al. Cham: Springer International Publishing, 2017, pp. 11–19. ISBN: 978-3-319-65849-0.

[86] Shifan Gao, Bing Chen, Yiming Qu, et al. "MRAM Acceleration Core for Vector Matrix Multiplication and XNOR-Binarized Neural Network Inference". In: *2020 International Symposium on VLSI Technology, Systems and Applications (VLSI-TSA)*. 2020, pp. 153–154. DOI: `10.1109/VLSI-TSA48913.2020.9203740`. URL: `https://ieeexplore.ieee.org/abstract/document/9203740`.

[87] Andreas Geiger, Philip Lenz, Christoph Stiller, et al. "Vision meets Robotics: The KITTI Dataset". In: *International Journal of Robotics Research (IJRR)* (2013).

[88] Andreas Geiger, Philip Lenz, and Raquel Urtasun. "Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2012.

[89] Amir Gholami. "AI and the Memory Wall". In: *Medium* (2023). Accessed: 14.12.2023. URL: `https://medium.com/riselab/ai-and-memory-wall-2cb4265cb0b8`.

[90] Davide Giri, Kuan-Lin Chiu, Guy Eichler, et al. "Accelerator Integration for Open-Source SoC Design". In: *IEEE Micro* 41.4 (2021), pp. 8–14. DOI: `10.1109/MM.2021.3073893`.

[91] Akhetonics GmbH. *Akhetonics - Homepage*. `https://www.akhetonics.com/`. Accessed: on 08.02.2024. 2022.

[92] Laurence Goasduff. *Gartner Predicts Chip Shortages Will Drive 50% of the Top 10 Automotive OEMs to Design Their Own Chips by 2025*. `https://www.gartner.com/en/newsroom/press-releases/2021-12-06-gartner-predicts-chip-shortages-will-drive-fifty-percent-of-the-top-10-automotive-oems-to-design-their-own-chips-by-2025`. Accessed: 13.02.2024. Dec. 2021.

[93] Alexandre Gonfalonieri. "How Amazon Alexa Works: Your Guide to Natural Language Processing AI". In: *Towards Data Science* (2018). Accessed: 10.02.2023. URL: `https://towardsdatascience.com/how-amazon-alexa-works-your-guide-to-natural-language-processing-ai-7506004709d3`.

[94] BMW Group. *BMW Group and Meta's Reality Labs present joint research for interlinking extended reality devices with the digital vehicle ecosystem*. BMW Group Press Release. Accessed: 12.12.2023. Mar. 2023. URL: `https://www.press.bmwgroup.com/global/article/detail/T0417479EN/bmw-group-and-meta%E2%80%99s-reality-labs-present-joint-research-for-interlinking-extended-reality-devices-with-the-digital-vehicle-ecosystem?language=en`.

[95] Yaliang Gui, Behrouz Movahhed Nouri, Mario Miscuglio, et al. "100 GHz micrometer-compact broadband monolithic ITO Mach–Zehnder interferometer modulator enabling 3500 times higher packing density". In: *Nanophotonics* 11.17 (Apr. 2022), 4001–4009. ISSN: 2192-8614. DOI: 10.1515/nanoph-2021-0796. URL: http://dx.doi.org/10.1515/nanoph-2021-0796.

[96] Jian Han, Yaping Liao, Junyou Zhang, et al. "Target Fusion Detection of LiDAR and Camera Based on the Improved YOLO Algorithm". In: *Mathematics* 6.10 (2018). ISSN: 2227-7390. DOI: 10.3390/math6100213. URL: https://www.mdpi.com/2227-7390/6/10/213.

[97] Nicholas Harris. "Introducing Envise, Idiom, and Passage: Next-Generation AI Compute, Compile, and Interconnect". In: *Medium* (2023). Accessed: 14.12.2023. URL: https://medium.com/lightmatter/introducing-envise-idiom-and-passage-next-generation-ai-compute-compile-and-interconnect-331878d6cea5.

[98] Anders Hast. "Improved Algorithms for Fast Shading and Lighting". PhD thesis. Acta Universitatis Upsaliensis, 2004.

[99] Kaiming He, Xiangyu Zhang, Shaoqing Ren, et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[100] John L Hennessy and David A Patterson. *Computer architecture: a quantitative approach*. Elsevier, 2011. URL: http://acs.pub.ro/~cpop/SMPA/Computer%20Architecture%20A%20Quantitative%20Approach%20(5th%20edition).pdf.

[101] Liu Hongbin, Hu Ren, Chen Wang, et al. "Towards a Peta-scale Unstructured Computational Fluid Dynamics (CFD) Acceleration Toolkit based on Sunway architectures?" In: Sept. 2019. DOI: 10.33737/gpps19-bj-148.

[102] Gerlas van den Hoven. "FTTH deployment taking off in Europe". In: *2008 5th International Conference on Broadband Communications, Networks and Systems*. 2008, pp. 221–221. DOI: 10.1109/BROADNETS.2008.4769074.

[103] Miao Hu, John Paul Strachan, Zhiyong Li, et al. "Dot-Product Engine for Neuromorphic Computing: Programming 1T1M Crossbar to Accelerate Matrix-Vector Multiplication". In: *Proceedings of the 53rd Annual Design Automation Conference*. DAC '16. Association for Computing Machinery, 2016. ISBN: 9781450342360. DOI: 10.1145/2897937.2898010. URL: https://doi.org/10.1145/2897937.2898010.

[104] Hao Huang, Johannes Heilmeyer, Markus Grözing, et al. "An 8-bit 100-GS/s distributed DAC in 28-nm CMOS". In: *2014 IEEE Radio Frequency Integrated Circuits Symposium*. 2014, pp. 65–68. DOI: 10.1109/RFIC.2014.6851659.

[105] Zhiyu Huang, Chen Lv, Yang Xing, et al. “Multi-Modal Sensor Fusion-Based Deep Neural Network for End-to-End Autonomous Driving With Scene Understanding”. In: *IEEE Sensors Journal* 21.10 (May 2021), 11781–11790. ISSN: 2379-9153. DOI: 10.1109/jsen.2020.3003121. URL: http://dx.doi.org/10.1109/JSEN.2020.3003121.

[106] Jonathan Hui. *AI Chips: Technology Trends & Landscape – Mobile SoC, Intel, Asian AI chips, Low Power Inference*. https://jonathan-hui.medium.com/ai-chips-technology-trends-landscape-mobile-soc-intel-asian-ai-chips-low-power-inference-4db701dbe85d. Accessed: 05.02.2024. 2021.

[107] IKA. *A test procedure for airbags - IKA Report 8328*. https://citainsp.org/wp-content/uploads/2016/01/ECS-RSP-Study-2-TP-airbags.pdf. Accessed: 04.12.2023. 2016.

[108] IMARC Group. *System-on-Chip (SoC) Market Size, Share, Trends, Forecast 2024-2032*. https://www.imarcgroup.com/system-on-chip-market. Accessed: 30.01.2024. 2024.

[109] Meta Inc. *Unwrap mixed reality with Meta Quest 3*. Accessed: 14.12.2023. 2023. URL: https://www.meta.com/de/en/quest/quest-3/.

[110] Ouster Inc. *Ouster OS1 Lidar Sensor Datasheet*. https://data.ouster.io/downloads/datasheets/datasheet-rev7-v3p0-os1.pdf. Accessed: 05.12.2023. 2023.

[111] Tesla Inc. *Tesla Model S*. Accessed: 14.12.2023. 2023. URL: https://www.tesla.com/models#:~:text=Cinematic%20Experience.

[112] Intel Corporation. *Pentium III Processor for the PGA370 Socket at 500 MHz to 1.13 GHz*. Accessed: 20.01.2024. 2001. URL: https://download.intel.com/design/PentiumIII/datashts/24526408.pdf.

[113] *Introduction to Cloud TPU*. https://cloud.google.com/tpu/docs/intro-to-tpu. Accessed: February 7, 2024. 2024.

[114] Valentin Isaac–Chassande, Adrian Evans, Yves Durand, et al. “Dedicated Hardware Accelerators for Processing of Sparse Matrices and Vectors: A Survey”. In: *ACM Trans. Archit. Code Optim.* (Jan. 2024). Just Accepted. ISSN: 1544-3566. DOI: 10.1145/3640542. URL: https://doi.org/10.1145/3640542.

[115] Joel Janai, Fatma Güney, Aseem Behl, et al. *Computer Vision for Autonomous Vehicles: Problems, Datasets and State of the Art*. 2021. arXiv: 1704.05519 [cs.CV].

[116] Aashu Jha, Chaoran Huang, and Paul R. Prucnal. “Reconfigurable all-optical nonlinear activation functions for neuromorphic photonics”. In: *Opt. Lett.* 45.17 (Sept. 2020), pp. 4819–4822. DOI: 10.1364/OL.398234. URL: https://opg.optica.org/ol/abstract.cfm?URI=ol-45-17-4819.

[117] Yue Jiang, Wenjia Zhang, Fan Yang, et al. “Photonic Convolution Neural Network Based on Interleaved Time-Wavelength Modulation”. In: *Journal of Lightwave Technology* 39.14 (2021), pp. 4592–4600. DOI: 10.1109/JLT.2021.3076070.

[118] Glenn Jocher and Sergiu Waxmann. *YOLOv5 Architecture Summary*. Ultralytics YOLOv8 Documentation. Accessed: 04.02.2024. Jan. 2024. URL: https://docs.ultralytics.com/yolov5/tutorials/architecture_description/?h=architec.

[119] Norman P. Jouppi, George Kurian, Sheng Li, et al. *TPU v4: An Optically Reconfigurable Supercomputer for Machine Learning with Hardware Support for Embeddings*. 2023. arXiv: 2304.01433 [cs.AR].

[120] Jørgensen, A.A., Kong, et al. “Petabit-per-second data transmission using a chip-scale microcomb ring resonator source”. In: *Nature Photonics 16* (2022). DOI: https://doi.org/10.1038/s41566-022-01082-z.

[121] Christoforos Kachris and Ioannis Tomkos. “A Survey on Optical Interconnects for Data Centers”. In: *IEEE Communications Surveys & Tutorials* 14.4 (2012), pp. 1021–1036. DOI: 10.1109/SURV.2011.122111.00069.

[122] David Kasperek, Pawel Antonowicz, Marek Baranowski, et al. “Comparison of the Usability of Apple M2 and M1 Processors for Various Machine Learning Tasks”. In: *Sensors* 23.12 (2023). DOI: 10.3390/s23125424. URL: https://www.mdpi.com/1424-8220/23/12/5424.

[123] Narsimlu Kemsaram, Anweshan Das, and Gijs Dubbelman. “A Stereo Perception Framework for Autonomous Vehicles”. In: *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*. 2020, pp. 1–6. DOI: 10.1109/VTC2020-Spring48590.2020.9128899.

[124] Salman Khan, Muzammal Naseer, Munawar Hayat, et al. “Transformers in Vision: A Survey”. In: *ACM Computing Surveys* 54.10s (Jan. 2022), 1–41. ISSN: 1557-7341. DOI: 10.1145/3505244. URL: http://dx.doi.org/10.1145/3505244.

[125] Jinsoo Kim, Jongwon Kim, and Jeongho Cho. “An advanced object classification strategy using YOLO through camera and LiDAR sensor fusion”. In: *2019 13th International Conference on Signal Processing and Communication Systems (ICSPCS)*. 2019, pp. 1–5. DOI: 10.1109/ICSPCS47537.2019.9008742.

[126] ZuWhan Kim. “Robust Lane Detection and Tracking in Challenging Scenarios”. In: *IEEE Transactions on Intelligent Transportation Systems* 9.1 (2008), pp. 16–26. DOI: 10.1109/TITS.2007.908582.

[127] Kingston Technology Company. *PCIe Gen 4 Explained*. https://www.kingston.com/en/blog/pc-performance/pcie-gen-4-explained. Accessed: 05.02.2024. 2021.

[128] Ken-ichi Kitayama, Masaya Notomi, Makoto Naruse, et al. “Novel frontier of photonics for data processing—Photonic accelerator”. In: *APL Photonics* 4.9 (Sept. 2019), p. 090901. ISSN: 2378-0967. DOI: 10.1063/1.5108912. URL: http://dx.doi.org/10.1063/1.5108912.

[129] *KITTI 3D Object Detection Benchmark*. https://www.cvlibs.net/datasets/kitti/eval_object.php?obj_benchmark=3d. Accessed: 05.02.2024.

[130] Knut and Alice Wallenberg Foundation. *Coupling Light and Magnetism on the Nanoscale*. Accessed: 02.02.2024. 2024. URL: https://kaw.wallenberg.org/en/research/coupling-light-magnetism-nanoscale.

[131] Jarno Laaksonen. “OpenGL rendering pipeline”. In: (2017).

[132] A. Lagae, S. Lefebvre, R. Cook, et al. “A Survey of Procedural Noise Functions”. In: *Computer Graphics Forum* 29.8 (Oct. 2010), 2579–2600. ISSN: 1467-8659. DOI: 10.1111/j.1467-8659.2010.01827.x. URL: http://dx.doi.org/10.1111/j.1467-8659.2010.01827.x.

[133] Alex H. Lang, Sourabh Vora, Holger Caesar, et al. *PointPillars: Fast Encoders for Object Detection from Point Clouds*. 2019. arXiv: 1812.05784 [cs.LG].

[134] Hyung-Jin Lee, Ravi Mahajan, Farhana Sheikh, et al. “Multi-die Integration Using Advanced Packaging Technologies”. In: *2020 IEEE Custom Integrated Circuits Conference (CICC)*. 2020, pp. 1–7. DOI: 10.1109/CICC48029.2020.9075901.

[135] Charles E. Leiserson, Neil C. Thompson, Joel S. Emer, et al. “There’s plenty of room at the Top: What will drive computer performance after Moore’s law?” In: *Science* 368.6495 (2020), eaam9744. DOI: 10.1126/science.aam9744. eprint: https://www.science.org/doi/pdf/10.1126/science.aam9744. URL: https://www.science.org/doi/abs/10.1126/science.aam9744.

[136] Eric Leong. “Bridging the Gap Between Modular and End-to-end Autonomous Driving Systems”. In: (2022).

[137] D.I. Lewin. “DNA computing”. In: *Computing in Science & Engineering* 4.3 (2002), pp. 5–8. DOI: 10.1109/5992.998634.

[138] Chong Li, Xiang Zhang, Jingwei Li, et al. “The challenges of modern computing and new opportunities for optics”. In: *PhotoniX* 2.1 (Sept. 2021). ISSN: 2662-1991. DOI: 10.1186/s43074-021-00042-0. URL: http://dx.doi.org/10.1186/s43074-021-00042-0.

[139] Xin Li, Tao Ma, Yuenan Hou, et al. *LoGoNet: Towards Accurate 3D Object Detection with Local-to-Global Cross-Modal Fusion*. 2023. arXiv: 2303.03595 [cs.CV].

[140] Zhichao Li, Feng Wang, and Naiyan Wang. *LiDAR R-CNN: An Efficient and Universal 3D Object Detector*. 2021. arXiv: 2103.15297 [cs.CV].

[141] Tingting Liang, Hongwei Xie, Kaicheng Yu, et al. *BEVFusion: A Simple and Robust LiDAR-Camera Fusion Framework*. 2022. arXiv: 2205.13790 [cs.CV].

[142] Lightelligence. *Large-Scale Optoelectronic Integration for Intelligent Computing Power Networks*. https://www.lightelligence.ai/bocupload/2023/04/24/large-scale-optoelectronic-integration-intelligent-computing-power-networks.pdf. 2023.

[143] Lightelligence. *Photonic Arithmetic Computing Engine (PACE)*. https://www.lightelligence.ai/index.php/product/index/2.html. Accessed: 08.02.2024. 2024.

[144] Lightmatter. *Envise*. https://lightmatter.co/products/envise/. Accessed: 31.12.2023. 2023.

[145] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, et al. *Feature Pyramid Networks for Object Detection*. 2017. arXiv: 1612.03144 [cs.CV].

[146] Xing Lin, Yair Rivenson, Nezih T. Yardimci, et al. "All-optical machine learning using diffractive deep neural networks". In: *Science* 361.6406 (2018), pp. 1004–1008. DOI: 10.1126/science.aat8084. eprint: https://www.science.org/doi/pdf/10.1126/science.aat8084. URL: https://www.science.org/doi/abs/10.1126/science.aat8084.

[147] Ying-Cheng Lin, Ping-Yen Chiang, and Shaou-Gang Miaou. "Enhancing Deep-Learning Object Detection Performance Based on Fusion of Infrared and Visible Images in Advanced Driver Assistance Systems". In: *IEEE Access* 10 (2022), pp. 105214–105231. DOI: 10.1109/ACCESS.2022.3211267.

[148] Brad Linder. *Intel's Unpopular Lakefield Chips Reach End of Life a Year After Launch*. https://liliputing.com/intels-unpopular-lakefield-chips-reach-end-of-life-a-year-after-launch/. Accessed: 30.01.2024. 2021.

[149] Qiaolv Ling, Penghui Dong, Yayan Chu, et al. "On-chip optical matrix-vector multiplier based on mode division multiplexing". In: *Chip* 2.4 (2023), p. 100061. ISSN: 2709-4723. DOI: https://doi.org/10.1016/j.chip.2023.100061. URL: https://www.sciencedirect.com/science/article/pii/S2709472323000242.

[150] Wei Liu, Dragomir Anguelov, Dumitru Erhan, et al. "SSD: Single Shot MultiBox Detector". In: *Computer Vision – ECCV 2016*. Ed. by Bastian Leibe, Jiri Matas, Nicu Sebe, et al. Cham: Springer International Publishing, 2016, pp. 21–37. ISBN: 978-3-319-46448-0.

[151] ARM Ltd. *Introduction To AMBA*. https://documentation-service.arm.com/static/5f106ce80daa596235e81425. [Online; accessed 2024-02-05]. ARM Ltd., 1996.

[152] Yujia Luo. *Time Constraints and Fault Tolerance in Autonomous Driving Systems*. 2019. URL: https://www2.eecs.berkeley.edu/Pubs/TechRpts/2019/EECS-2019-39.pdf.

[153] Jiawen Lv, Tianhang Lian, Baizhu Lin, et al. "Low power consumption mode switch based on three-dimensional polymer M-Z interferometer". In: *IEEE Photonics Journal* PP (May 2020), pp. 1–1. DOI: 10.1109/JPHOT.2020.2992745.

[154] MacRumors. "Apple to Buy TSMC's Entire Supply of 3nm Chips for 2023". In: (2023). Accessed: 30.01.2024. URL: https://www.macrumors.com/2023/08/29/apple-buys-all-3nm-tsmc-chips-2023/.

[155] Simone Mangiante, Guenter Klas, Amit Navon, et al. "VR is on the Edge: How to Deliver 360° Videos in Mobile Networks". In: Accessed: 07.12.2023. Aug. 2017, pp. 30–35. DOI: 10.1145/3097895.3097901.

[156] Dylan Martin. *Nvidia, Intel and others pour $130m into Ayar optical chips*. https://www.theregister.com/2022/04/26/nvidia_intel_ayar/. Accessed: 11.02.2024. 2022.

[157] A. Matthey, F. Rupalla, T. Schneiderbauer, et al. "How do consumers perceive in-car connectivity and digital services?" In: *McKinsey & Company, Automotive & Assembly Practice* (2023). Accessed: 20.12.2023.

[158] J.C. McCall and M.M. Trivedi. "An integrated, robust approach to lane marking detection and lane tracking". In: *IEEE Intelligent Vehicles Symposium, 2004*. 2004, pp. 533–537. DOI: 10.1109/IVS.2004.1336440.

[159] Peter L. McMahon. *The physics of optical computing*. 2023. arXiv: 2308.00088 [physics.optics].

[160] Akshit Mehra. "Understanding YOLOv8 Architecture, Applications & Features". In: *Labellerr Blog* (Apr. 2023). URL: https://www.labellerr.com/blog/understanding-yolov8-architecture-applications-features/.

[161] Maria I. Mera Collantes and Siddharth Garg. "Do Not Trust, Verify: A Verifiable Hardware Accelerator for Matrix Multiplication". In: *IEEE Embedded Systems Letters* 12.3 (2020), pp. 70–73. DOI: 10.1109/LES.2019.2953485.

[162] Microsoft. *Xbox Cloud Gaming (Beta) | Xbox — xbox.com*. https://www.xbox.com/en-US/cloud-gaming. Accessed 08.02.2024. 2023.

[163] Sparsh Mittal and Jeffrey S. Vetter. "A Survey of CPU-GPU Heterogeneous Computing Techniques". In: *ACM Comput. Surv.* 47.4 (July 2015). ISSN: 0360-0300. DOI: 10.1145/2788396. URL: https://doi.org/10.1145/2788396.

[164] SHIHO MIYAJIMA. *Japan's NTT, Intel to collaborate on cutting-edge chips using optical tech.* Accessed: 11.02.2024. 2024. URL: `https://asia.nikkei.com/Business/Tech/Semiconductors/Japan-s-NTT-Intel-to-collaborate-on-cutting-edge-chips-using-optical-tech`.

[165] B. Mochocki, K. Lahiri, and S. Cadambi. "Power Analysis of Mobile 3D Graphics". In: *Proceedings of the Design Automation & Test in Europe Conference*. Vol. 1. 2006, pp. 1–6. DOI: `10.1109/DATE.2006.243859`.

[166] B. Moons. *Transformers in Computer Vision*. `https://www.edge-ai-vision.com/2022/05/transformers-in-computer-vision/`. Accessed: 23.01.2023. May 2022.

[167] Gordon E Moore et al. *Moore's law at 40*. 2006.

[168] G. Mourgias-Alexandris, M. Moralis-Pegios, A. Tsakyridis, et al. "Noise-resilient and high-speed deep learning with coherent silicon photonics". In: *Nature Communications* 13.1 (Sept. 2022). ISSN: 2041-1723. DOI: `10.1038/s41467-022-33259-z`. URL: `http://dx.doi.org/10.1038/s41467-022-33259-z`.

[169] Agus Mulyanto, Rohmat Indra Borman, Purwono Prasetyawan, et al. "Implementation 2D Lidar and Camera for detection object and distance based on RoS". In: *JOIV: International Journal on Informatics Visualization* 4.4 (Dec. 2020), 231–236. ISSN: 2549-9610. DOI: `10.30630/joiv.4.4.466`. URL: `http://dx.doi.org/10.30630/joiv.4.4.466`.

[170] Jamuna S Murthy, GM Siddesh, Wen-Cheng Lai, et al. "Objectdetect: A real-time object detection framework for advanced driver assistant systems using yolov5". In: *Wireless Communications and Mobile Computing* 2022 (2022).

[171] Pedro Javier Navarro, Francisca Rosique, Carlos Fernández, et al. *Chapter 7 - End-to-end architectures*. Ed. by Jorge Villagra and Felipe Jiménez. Elsevier, 2023, pp. 169–192. ISBN: 978-0-323-98339-6. DOI: `https://doi.org/10.1016/B978-0-323-98339-6.00009-9`. URL: `https://www.sciencedirect.com/science/article/pii/B9780323983396000099`.

[172] Mixed News. *Apple Vision Pro Displays Cost*. Accessed: 14.12.2023. 2023.

[173] NIO. *NOMI: The World's First In-Vehicle Artificial Intelligence*. NIO Official Blog. Accessed: 12.12.2023. Feb. 2023. URL: `https://www.nio.com/blog/nomi-worlds-first-vehicle-artificial-intelligence`.

[174] NVIDIA. *Achieving FP32 Accuracy for INT8 Inference Using Quantization Aware Training with TensorRT*. `https://developer.nvidia.com/blog/achieving-fp32-accuracy-for-int8-inference-using-quantization-aware-training-with-tensorrt/`. Accessed: 31.01.2024. 2020.

[175] NVIDIA. *NVIDIA DRIVE Hyperion Autonomous Vehicle Reference Architecture*. Accessed: 06.12.2023. URL: `https://developer.nvidia.com/drive/hyperion`.

[176] NVIDIA. *NVIDIA DRIVE Partner Ecosystem.* Accessed: 06.12.2023. URL: https://www.nvidia.com/en-us/self-driving-cars/partners/.

[177] Shuhei Ohno, Rui Tang, Kasidit Toprasertpong, et al. "Si Microring Resonator Crossbar Array for On-Chip Inference and Training of the Optical Neural Network". In: *ACS Photonics* 9.8 (2022), pp. 2614–2622. DOI: 10.1021/acsphotonics.1c01777. eprint: https://doi.org/10.1021/acsphotonics.1c01777. URL: https://doi.org/10.1021/acsphotonics.1c01777.

[178] Shuhei Ohno, Kasidit Toprasertpong, Shinichi Takagi, et al. *Si microring resonator crossbar array for on-chip inference and training of optical neural network.* 2021. arXiv: 2106.04351 [cs.ET].

[179] Jun Rong Ong, Chin Chun Ooi, Thomas Y. L. Ang, et al. "Photonic Convolutional Neural Networks Using Integrated Diffractive Optics". In: *IEEE Journal of Selected Topics in Quantum Electronics* 26.5 (2020), pp. 1–8. DOI: 10.1109/JSTQE.2020.2982990.

[180] David Oppenheimer, Archana Ganapathi, and David A Patterson. "Why do Internet services fail, and what can be done about it?" In: *4th Usenix Symposium on Internet Technologies and Systems (USITS 03)*. 2003.

[181] Optalysys. "Optalysys: What we've done (And why we did it)". In: *Medium* (2022). Accessed: 26.12.2023. URL: https://medium.com/optalysys/optalysys-what-weve-done-and-why-we-did-it-f01591374cb6.

[182] Shaoyuan Ou, Alexander Sludds, Ryan Hamerly, et al. *Hypermultiplexed Integrated Tensor Optical Processor*. 2024. arXiv: 2401.18050 [cs.ET].

[183] Jaehyoung Park, Geun Young Kim, Hyung Jin Park, et al. "FTTH Deployment Status & Strategy in Korea: GW-PON Based FTTH Field Trial and Reach Extension Strategy of FTTH in Korea". In: *IEEE GLOBECOM 2008 - 2008 IEEE Global Telecommunications Conference*. 2008, pp. 1–3. DOI: 10.1109/GLOCOM.2008.ECP.1074.

[184] Sunghyun Park, Tushar Krishna, Chia-Hsin Chen, et al. "Approaching the theoretical limits of a mesh NoC with a 16-node chip prototype in 45nm SOI". In: *DAC Design Automation Conference 2012*. 2012, pp. 398–405.

[185] Nikolaos Passalis, George Mourgias-Alexandris, Apostolos Tsakyridis, et al. "Training Deep Photonic Convolutional Neural Networks With Sinusoidal Activations". In: *IEEE Transactions on Emerging Topics in Computational Intelligence* 5.3 (2021), pp. 384–393. DOI: 10.1109/TETCI.2019.2923001.

[186] Dylan Patel. *Google New Custom Silicon replaces 10 million Intel cpus: Google Argos Vpu*. June 2021. URL: https://www.semianalysis.com/p/google-new-custom-silicon-replaces.

[187] Scott Drew Pendleton, Hans Andersen, Xinxin Du, et al. "Perception, Planning, Control, and Coordination for Autonomous Vehicles". In: *Machines* 5.1 (2017). Accessed: 01.01.2024. ISSN: 2075-1702. DOI: 10.3390/machines5010006. URL: https://www.mdpi.com/2075-1702/5/1/6.

[188] Peripheral Component Interconnect Special Interest Group (PCI-SIG). *PCI Express 6.0 Specification*. Accessed: 05.02.2024. 2024. URL: https://pcisig.com/pci-express-6.0-specification.

[189] Michele Petracca, Benjamin G. Lee, Keren Bergman, et al. "Photonic NoCs: System-Level Design Exploration". In: *IEEE Micro* 29.4 (2009), pp. 74–85. DOI: 10.1109/MM.2009.70.

[190] Jin-Chun Piao and Shin-Dug Kim. "Adaptive Monocular Visual–Inertial SLAM for Real-Time Augmented Reality Applications in Mobile Devices". In: *Sensors* 17.11 (2017). ISSN: 1424-8220. DOI: 10.3390/s17112567. URL: https://www.mdpi.com/1424-8220/17/11/2567.

[191] James R. Powell. "The Quantum Limit to Moore's Law". In: *Proceedings of the IEEE* 96.8 (2008), pp. 1247–1248. DOI: 10.1109/JPROC.2008.925411.

[192] BMW Group PressClub. *BMW Theatre Screen brings cinematic experience into the vehicle*. Accessed: 14.12.2023. 2021. URL: https://www.press.bmwgroup.com/global/article/detail/T0363131EN/bmw-theatre-screen-brings-cinematic-experience-into-the-vehicle?language=en.

[193] BMW Group PressClub. *MINI Presents Spike, the Digital Character for the New Model Family*. Accessed: 12.12.2023. Mar. 2021. URL: https://www.press.bmwgroup.com/global/article/detail/T0412718EN/mini-presents-spike-the-digital-character-for-the-new-model-family?language=en.

[194] BMW Group PressClub. *The all-new BMW iDrive*. Accessed: 12.12.2023. Mar. 2021. URL: https://www.press.bmwgroup.com/global/article/detail/T0327315EN/the-all-new-bmw-idrive.

[195] J. Schacht i q. *Funktionale Sicherheit (FuSi) – die ASIL-Klassifikation ...* Accessed: 06.12.2023. 2016. URL: https://www.i-q.de/iso-26262/fusi-asil-klassifikationen.

[196] Charles R. Qi, Hao Su, Kaichun Mo, et al. *PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation*. 2017. arXiv: 1612.00593 [cs.CV].

[197] T. Qin and Amazon Science S. Dayal. *Low-precision arithmetic makes robot localization more efficient*. Accessed: 31.12.2023. 2023. URL: https://www.amazon.science/blog/low-precision-arithmetic-makes-robot-localization-more-efficient.

[198] Yuhao Qing, Wenyi Liu, Liuyan Feng, et al. "Improved YOLO Network for Free-Angle Remote Sensing Target Detection". In: *Remote Sensing* 13.11 (June 2021), p. 2171. ISSN: 2072-4292. DOI: 10.3390/rs13112171. URL: http://dx.doi.org/10.3390/rs13112171.

[199] Qualcomm Technologies, Inc. *Qualcomm Snapdragon 888 5G Mobile Platform Product Brief*. https://www.qualcomm.com/content/dam/qcomm-martech/dm-assets/documents/prod_brief_qcom_sd888_5g_0.pdf. Accessed: 30.01.2024. 2020.

[200] Qualcomm Technologies, Inc. *Samsung Galaxy S21 5G Featuring Qualcomm Snapdragon Processor*. https://www.qualcomm.com/snapdragon/device-finder/samsung-galaxy-s21-5g. Accessed: 30.01.2024. 2021.

[201] Pushkar Ranade. *How the SoC is Displacing the CPU*. https://medium.com/@magicsilicon/how-the-soc-is-displacing-the-cpu-49bc7503edab. Accessed: 10.02.2024. 2015.

[202] Sabbir Rangwala. *Innoviz Scores a Major Lidar Win with a Dominant Automotive OEMs & A Tier-1 Supplier*. Accessed: 05.12.2023. 2022. URL: https://www.forbes.com/sites/sabbirrangwala/2022/08/02/innoviz-scores-a-major-lidar-win-with-a-dominant-automotive-oemas-a-tier-1-supplier/?sh=43a901a4221d.

[203] P. Rawat. *Environment Perception for Autonomous Driving : A 1/10 Scale Implementation Of Low Level Sensor Fusion Using Occupancy Grid Mapping*. 2019.

[204] Joseph Redmon, Santosh Divvala, Ross Girshick, et al. *You Only Look Once: Unified, Real-Time Object Detection*. 2016. arXiv: 1506.02640 [cs.CV].

[205] Joseph Redmon and Ali Farhadi. *YOLO9000: Better, Faster, Stronger*. 2016. arXiv: 1612.08242 [cs.CV].

[206] Dominik Reinhardt, Udo Dannebaum, Michael Scheffer, et al. "High Performance Processor Architecture for Automotive Large Scaled Integrated Systems within the European Processor Initiative Research Project". In: Apr. 2019. DOI: 10.4271/2019-01-0118.

[207] Shaoqing Ren, Kaiming He, Ross Girshick, et al. *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*. 2016. arXiv: 1506.01497 [cs.CV].

[208] *Ultimate 0.34 nm Gate-length Side-Wall Transistors with Atomic Level Channel*. Dec. 2020. DOI: 10.21203/rs.3.rs-119491/v1.

[209] A.M. Rincon, W.R. Lee, and M. Slattery. "The changing landscape of system-on-a-chip design". In: *Proceedings of the IEEE 1999 Custom Integrated Circuits Conference (Cat. No.99CH36327)*. 1999, pp. 83–90. DOI: 10.1109/CICC.1999.777248.

[210] Jesse Roch, Jamil Fayyad, and Homayoun Najjaran. "DOPESLAM: High-Precision ROS-Based Semantic 3D SLAM in a Dynamic Environment". In: *Sensors (Basel)* 23.9 (Apr. 2023).

[211] Max Roser. *Moore's Law*. Our World in Data. Accessed: 04.01.2024. 2021. URL: https://ourworldindata.org/moores-law.

[212] Max Roser. "The brief history of artificial intelligence: The world has changed fast – what might be next?" In: *Our World in Data* (2022). Accessed: 10.02.2023. URL: https://ourworldindata.org/brief-history-of-ai.

[213] Matt Ross. "Under the Hood of Neural Network Forward Propagation: The Dreaded Matrix Multiplication". In: *Towards Data Science* (2017). Accessed: 19.01.2024. URL: https://towardsdatascience.com/under-the-hood-of-neural-network-forward-propagation-the-dreaded-matrix-multiplication-a5360b33426.

[214] M. Rotundo, A. Leoni, L. Serafini, et al. *Simulation and Validation of a SpaceWire On-Board Data-Handling Network for the PLATO Mission*. 2018. arXiv: 1808.09874 [astro-ph.IM].

[215] Thales Luis Sabino, Paulo Andrade, Esteban Walter Gonzales Clua, et al. "A hybrid GPU rasterized and ray traced rendering pipeline for real time rendering of per pixel effects". In: *Entertainment Computing-ICEC 2012: 11th International Conference, ICEC 2012, Bremen, Germany, September 26-29, 2012. Proceedings 11*. Springer. 2012, pp. 292–305.

[216] Sumit Saha, Tarek Naous, and Zuraiz Ahmed. "A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way". In: *Towards Data Science* (2018). Accessed: 18.01.2024. URL: https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53.

[217] Andrey Sakryukin. *Under The Hood of Neural Networks. Part 1: Fully Connected. | by Andrey Sakryukin | Towards Data Science*. https://towardsdatascience.com/under-the-hood-of-neural-networks-part-1-fully-connected-5223b7f78528. Accessed: 11.02.2024. Apr. 2018.

[218] Kaz Sato and Cliff Young. *An in-depth look at Google's first Tensor Processing Unit (TPU)*. Accessed: 03.02.2024. 2017. URL: https://cloud.google.com/blog/products/ai-machine-learning/an-in-depth-look-at-googles-first-tensor-processing-unit-tpu?hl=en.

[219] Patrick R. Schaumont. "System on Chip". In: *A Practical Introduction to Hardware/Software Codesign*. Boston, MA: Springer US, 2013, pp. 237–265. ISBN: 978-1-4614-3737-6. DOI: 10.1007/978-1-4614-3737-6_8. URL: https://doi.org/10.1007/978-1-4614-3737-6_8.

[220] Harold Serrano. *Noise in Computer Graphics- A brief introduction*. Accessed: 19.01.2024. 2015. URL: https://www.haroldserrano.com/blog/noise-in-computer-graphics-a-brief-introduction.

[221] Alok Sethi, Janne P. Aikio, Rana A. Shaheen, et al. “A 10-bit active RF phase shifter for 5G wireless systems”. In: *2017 IEEE Nordic Circuits and Systems Conference (NORCAS): NORCHIP and International Symposium of System-on-Chip (SoC)*. 2017, pp. 1–4. DOI: 10.1109/NORCHIP.2017.8124958.

[222] Jaime Sevilla, Lennart Heim, Anson Ho, et al. “Compute Trends Across Three Eras of Machine Learning”. In: *2022 International Joint Conference on Neural Networks (IJCNN)*. IEEE, July 2022. DOI: 10.1109/ijcnn55064.2022.9891914. URL: http://dx.doi.org/10.1109/IJCNN55064.2022.9891914.

[223] Amin Shafiee, Sanmitra Banerjee, Krishnendu Chakrabarty, et al. *Analysis of Optical Loss and Crosstalk Noise in MZI-based Coherent Photonic Neural Networks*. 2023. arXiv: 2308.03249 [cs.ET].

[224] John Shalf. “The future of computing beyond Moore’s Law”. In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 378 (Jan. 2020), p. 20190061. DOI: 10.1098/rsta.2019.0061.

[225] Ankit Sharma. “Evaluation of AXI-Interfaces for Hardware Software Communication”. PhD thesis. Feb. 2019.

[226] Yichen Shen, Nicholas C. Harris, Scott Skirlo, et al. “Deep learning with coherent nanophotonic circuits”. In: *2017 IEEE Photonics Society Summer Topical Meeting Series (SUM)*. 2017, pp. 189–190. DOI: 10.1109/PHOSST.2017.8012714.

[227] Yiting Shen and Wei Qi Yan. “Blind Spot Monitoring Using Deep Learning”. In: *2018 International Conference on Image and Vision Computing New Zealand (IVCNZ)*. 2018, pp. 1–5. DOI: 10.1109/IVCNZ.2018.8634716.

[228] Hualian Sheng, Sijia Cai, Yuan Liu, et al. *Improving 3D Object Detection with Channel-wise Transformer*. 2021. arXiv: 2108.10723 [cs.CV].

[229] anton Shilov. *Intel’s German fab will be most advanced in the world and make 1.5nm chips, CEO says | Tom’s Hardware*. https://www.tomshardware.com/tech-industry/manufacturing/intels-german-fab-will-be-most-advanced-in-the-world-and-make-15nm-chips-ceo-say. Accessed: 08.02.2024. Jan. 2024.

[230] Anton Shilov. “Micron Unveils HBM3 Gen2 Memory: 1.2 TB/sec Memory Stacks For HPC and AI Processors”. In: *AnandTech* (2023). URL: https://www.anandtech.com/show/18981/micron-unveils-hbm3-gen2-12-tbs-per-stack-at-92-gts-speed.

[231] Paul W. Shumate. “Fiber-to-the-Home: 1977–2007”. In: *Journal of Lightwave Technology* 26.9 (2008), pp. 1093–1103. DOI: 10.1109/JLT.2008.923601.

[232] Siemens Digital Industries Software. *Satisfy ISO 26262 safety requirements with builtin self-test*. TODO. Accessed: 30.11.2023. 2019. URL: https://resources.sw.siemens.com/de-DE/white-paper-using-built-in-self-test-hardware-to-satisfy-iso-26262-safety-requirements.

[233] Karen Simonyan and Andrew Zisserman. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2015. arXiv: 1409.1556 [cs.CV].

[234] Vishwanath A. Sindagi, Yin Zhou, and Oncel Tuzel. "MVX-Net: Multimodal VoxelNet for 3D Object Detection". In: *2019 International Conference on Robotics and Automation (ICRA)*. 2019, pp. 7276–7282. DOI: 10.1109/ICRA.2019.8794195.

[235] Apoorv Singh. *Transformer-Based Sensor Fusion for Autonomous Driving: A Survey*. 2023. arXiv: 2302.11481 [cs.CV].

[236] Ilamaran Sivarajah. *Lightmatter Propels Photonic Computing with The Envise Platform*. https://www.azooptics.com/Article.aspx?ArticleID=2371. Accessed: 07.02.2024. 2023.

[237] *SNAPDRAGON® 8 GEN 3 MOBILE PLATFORM*. Accessed: 04.02.2024. Qualcomm Technologies, Inc. 2023.

[238] James Spall, Xianxin Guo, Thomas D. Barrett, et al. "Fully reconfigurable coherent optical vector–matrix multiplication". In: *Optics Letters* 45.20 (2020). ISSN: 1539-4794. DOI: 10.1364/ol.401675. URL: http://dx.doi.org/10.1364/OL.401675.

[239] Josef Spjut, Andrew Kensler, and Erik Brunvand. "Hardware-accelerated gradient noise for graphics". In: May 2009, pp. 457–462. DOI: 10.1145/1531542.1531647.

[240] Starwin. *World's First Pre-installed Built-in ESA Terminal on Vehicle*. https://www.starwincom.com/newsinfo/871163.html. Accessed: 07.12.203. 2023.

[241] Stellar Telecommunications SAS. *Tour de France 2023: bad internet along the whole route*. https://www.stellar.tc/post/tour-de-france-2023-bad-internet-along-the-whole-route. Accessed: 07.12.2023. 2023.

[242] Stereolabs. *Performance Benchmark of YOLO v5, v7 and v8*. https://www.stereolabs.com/blog/performance-of-yolo-v5-v7-and-v8. Accessed: 02.01.2024. 2023.

[243] Gilbert Strang. "A proposal for Toeplitz matrix calculations". In: *Studies in Applied Mathematics* 74.2 (1986), pp. 171–176.

[244] Gilbert Strang. *Introduction to Linear Algebra*. 5th. Wellesley-Cambridge Press, 2016. ISBN: 978-0980232776.

[245] Yifan Sun, Nicolas Bohm Agostini, Shi Dong, et al. *Summarizing CPU and GPU Design Trends with Product Data*. 2020. arXiv: 1911.11313 [cs.DC].

[246] Hamed Taheri Gorji, Mojtaba Shahabi, Akshay Sharma, et al. “Combining deep learning and fluorescence imaging to automatically identify fecal contamination on meat carcasses”. In: *Scientific Reports* 12 (Feb. 2022), p. 2392. DOI: 10.1038/s41598-022-06379-1.

[247] Poohsan N Tamura and James C Wyant. “Matrix multiplication using coherent optical techniques”. In: *Optical Information Processing: Real Time Devices & Novel Techniques*. Vol. 83. SPIE. 1977, pp. 97–104.

[248] Mingxing Tan and Quoc V. Le. *EfficientNet: Improving Accuracy and Efficiency through AutoML and Model Scaling*. https://blog.research.google/2019/05/efficientnet-improving-accuracy-and.html. Accessed: 22.01.2024. 2019.

[249] Rui Tang, Ryota Tanomura, Takuo Tanemura, et al. “Ten-port unitary optical processor on a silicon photonic chip”. In: *ACS Photonics* 8.7 (2021), pp. 2074–2080.

[250] Ryota Tanomura, Rui Tang, Go Soma, et al. “All-Optical MIMO Demultiplexing Using Silicon-Photonic Dual-Polarization Optical Unitary Processor”. In: *Journal of Lightwave Technology* 41.12 (2023), pp. 3791–3796. DOI: 10.1109/JLT.2023.3276003.

[251] Ryota Tanomura, Rui Tang, Takuo Tanemura, et al. “Demonstration of Error-Tolerant Integrated Optical Processors Based on Multi-Plane Light Conversion”. In: *IEEE Photonics Technology Letters* 35.23 (2023), pp. 1275–1278. DOI: 10.1109/LPT.2023.3315781.

[252] Tomasz Tarnowski, Robert Haidenthaler, Matthias Pohl, et al. “Mercedes-Benz MBUX Hyperscreen Merges Technologies into Digital Dashboard Application”. In: *Information Display* 38.3 (2022), pp. 12–17. DOI: https://doi.org/10.1002/msid.1300. eprint: https://sid.onlinelibrary.wiley.com/doi/pdf/10.1002/msid.1300. URL: https://sid.onlinelibrary.wiley.com/doi/abs/10.1002/msid.1300.

[253] Brad Templeton. “Former Head Of Tesla AI Explains Why They’ve Removed Sensors, Others Differ”. In: *Forbes* (Oct. 2022). Accessed: 20.01.2024. URL: https://www.forbes.com/sites/bradtempleton/2022/10/31/former-head-of-tesla-ai-explains-why-theyve-removed-sensors-others-differ/?sh=5e304bf04ba8.

[254] Tesla. “Tesla, IEEE Hot Chips 31 Symposium (HCS) 2019”. In: *35th IEEE Hot Chips Symposium, HCS 2019*. IEEE. Palo Alto, CA, USA, 2019.

[255] @thegranturismo. *Vehicles from the first GranTurismo...* https://twitter.com/thegranturismo/status/1496490013248434176. Accessed: 07.01.2024. 2022.

[256] Ye Tian, Yang Zhao, Shengping Liu, et al. “Scalable and compact photonic neural chip with low learning-capability-loss”. In: *Nanophotonics* 11.2 (2022), pp. 329–344. DOI: doi:10.1515/nanoph-2021-0521. URL: https://doi.org/10.1515/nanoph-2021-0521.

[257] Binayak Tiwari, Mei Yang, Xiaohang Wang, et al. *In-Network Accumulation: Extending the Role of NoC for DNN Acceleration*. 2022. arXiv: 2209.10056 [cs.AR].

[258] Weiyu Tong, Yanxian Wei, Hailong Zhou, et al. “The Design of a Low-Loss, Fast-Response, Metal Thermo-Optic Phase Shifter Based on Coupled-Mode Theory”. In: *Photonics* 9.7 (2022). ISSN: 2304-6732. DOI: 10.3390/photonics9070447. URL: https://www.mdpi.com/2304-6732/9/7/447.

[259] Angelina Totovic, Christos Pappas, Manos Kirtas, et al. “WDM equipped universal linear optics for programmable neuromorphic photonic processors”. In: *Neuromorphic Computing and Engineering* 2.2 (June 2022), p. 024010. DOI: 10.1088/2634-4386/ac724d. URL: https://dx.doi.org/10.1088/2634-4386/ac724d.

[260] Charlotte Trueman. *Photonic computing company Lightmatter reaches unicorn status*. https://www.datacenterdynamics.com/en/news/photonic-computing-company-lightmatter-reaches-unicorn-status/. Accessed: 11.02.2024. 2023.

[261] Wen-Chung Tsai, Ying-Cherng Lan, Yu-Hen Hu, et al. “Networks on Chips: Structure and Design Methodologies”. In: *Journal of Electrical and Computer Engineering* 2012 (2012), 1–15. ISSN: 2090-0155. DOI: 10.1155/2012/509465. URL: http://dx.doi.org/10.1155/2012/509465.

[262] Tzu-Yun Tseng and Jian-Jiun Ding. “Vehicle Distance Estimation Method Based on Monocular Camera”. In: *2020 International Symposium on Computer, Consumer and Control (IS3C)*. 2020, pp. 102–105. DOI: 10.1109/IS3C50286.2020.00034.

[263] Aris Tsirigotis, George Sarantoglou, Stavros Deligiannidis, et al. *Integrated Photonic Accelerator Based on Optical Spectrum Slicing for Convolutional Neural Networks*. 2023. arXiv: 2303.10357 [cs.ET].

[264] Amin Vahdat and Mark Lohmeyer. *Enabling next-generation AI workloads: Announcing TPU v5p and AI Hypercomputer*. Accessed: 12.02.2024. Dec. 2023. URL: https://cloud.google.com/blog/products/ai-machine-learning/introducing-cloud-tpu-v5p-and-ai-hypercomputer?hl=en.

[265] Nitin Vaish. “Self-driving Cars and Power Consumption — New Chip Designs”. In: *Medium* (2023). Accessed: 20.01.2024. URL: https://nitinvaish.medium.com/self-driving-cars-and-power-consumption-new-chip-designs-4c723659f8cd.

[266] Ashish Vaswani, Noam Shazeer, Niki Parmar, et al. *Attention Is All You Need*. 2023. arXiv: 1706.03762 [cs.CL].

[267] Verdict. *Rapidus teams up with France’s Leti on 1-nm chip technology*. Accessed: 05.01.2024. 2023. URL: https://www.verdict.co.uk/rapidus-1-nm-chip-technology/?cf-view.

[268] Christoph Vetter, Christoph Guetter, Chenyang Xu, et al. “Non-rigid multi-modal registration on the GPU”. In: *Proc SPIE* (Mar. 2007).

[269] William J Vigrass. "Calculation of semiconductor failure rates". In: *Harris Semiconductor* (2010). URL: https://www.renesas.com/us/en/document/qsg/calculation-semiconductor-failure-rates.

[270] P. Gonzalez Vivo and J. Lowe. *Generative designs - The Book of Shaders*. Accessed: 06.01.2024. URL: https://thebookofshaders.com/10/.

[271] P. Gonzalez Vivo and J. Lowe. *Glossary of Terms - The Book of Shaders*. Accessed: 01.01.2024. URL: https://thebookofshaders.com/glossary/?search=vec3.

[272] Jian Wang and Yun Long. "On-chip silicon photonic signaling and processing: a review". In: *Science Bulletin* 63.19 (2018), pp. 1267–1310. ISSN: 2095-9273. DOI: https://doi.org/10.1016/j.scib.2018.05.038. URL: https://www.sciencedirect.com/science/article/pii/S209592731830327X.

[273] Kai Wang, Wei Li, Liu Yanjiang, et al. "A high-efficient and low-cost secure AMBA framework utilizing configurable data encryption modeling against probe attacks". In: *IEICE Electronics Express* 18 (Mar. 2021). DOI: 10.1587/elex.18.20210105.

[274] Ke Wang, Tianqiang Zhou, Zhichuang Zhang, et al. "PVF-DectNet: Multi-modal 3D detection network based on Perspective-Voxel fusion". In: *Engineering Applications of Artificial Intelligence* 120 (2023), p. 105951. ISSN: 0952-1976. DOI: https://doi.org/10.1016/j.engappai.2023.105951. URL: https://www.sciencedirect.com/science/article/pii/S0952197623001355.

[275] Mi Wang, Xiangfeng Chen, Umar Khan, et al. "Programmable wavelength filter with double ring loaded MZI". In: *Scientific Reports* 12.1 (Jan. 2022). ISSN: 2045-2322. DOI: 10.1038/s41598-021-04598-6. URL: http://dx.doi.org/10.1038/s41598-021-04598-6.

[276] Shaojie Wang, Tong Wu, and Yevgeniy Vorobeychik. "Towards Robust Sensor Fusion in Visual Perception". In: *CoRR* abs/2006.13192 (2020). arXiv: 2006.13192. URL: https://arxiv.org/abs/2006.13192.

[277] S. Ward-Foxton. "Mercedes Applies Neuromorphic Computing in EV Concept Car". In: *EE Times* (2023). Accessed: 17.01.2024. URL: https://www.eetimes.com/mercedes-applies-neuromorphic-computing-in-ev-concept-car/.

[278] Sally Ward-Foxton. "Lightelligence Debuts Electronic AI Accelerator With Optical NoC". In: *EE Times* (2023). Accessed: 14.12.2023. URL: https://www.eetimes.com/lightelligence-debuts-new-electronic-ai-accelerator/.

[279] Kyle Wiggers. *Photonics Startup Lightmatter Details P1, Its AI Optical Accelerator Chip*. Accessed: 03.02.2024. 2020. URL: https://venturebeat.com/ai/photonics-startup-lightmatter-details-p1-its-ai-optical-accelerator-chip.

[280] WikiChip. *FSD Chip - Tesla*. https://en.wikichip.org/wiki/tesla_(car_company)/fsd_chip. Accessed: 2024-02-04. WikiChip, 2023.

[281] WikiChip. *Tesla Full Self-Driving (FSD) Chip*. `https://en.wikichip.org/wiki/tesla_(car_company)/fsd_chip`. Accessed: 05.02.2024. 2023.

[282] R Stanley Williams. "What's Next?[The end of Moore's law]". In: *Computing in Science & Engineering* 19.2 (2017), pp. 7–13.

[283] Hai Wu, Jinhao Deng, Chenglu Wen, et al. "CasA: A Cascade Attention Network for 3-D Object Detection From LiDAR Point Clouds". In: *IEEE Transactions on Geoscience and Remote Sensing* 60 (2022), pp. 1–11. DOI: `10.1109/TGRS.2022.3203163`.

[284] Hai Wu, Chenglu Wen, Shaoshuai Shi, et al. "Virtual Sparse Convolution for Multimodal 3D Object Detection". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 21653–21662.

[285] Junyi Xin and Guanqun Sun. "Learn from Each Other: Comparison and Fusion for Medical Segmentation Loss". In: *2021 7th International Conference on Computer and Communications (ICCC)*. 2021, pp. 662–666. DOI: `10.1109/ICCC54389.2021.9674501`.

[286] Danfei Xu, Dragomir Anguelov, and Ashesh Jain. "PointFusion: Deep Sensor Fusion for 3D Bounding Box Estimation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2018.

[287] Yan Yan, Yuxing Mao, and Bo Li. "SECOND: Sparsely Embedded Convolutional Detection". In: *Sensors* 18.10 (2018). ISSN: 1424-8220. DOI: `10.3390/s18103337`. URL: `https://www.mdpi.com/1424-8220/18/10/3337`.

[288] De Jong Yeong, Gustavo Velasco-Hernandez, John Barry, et al. "Sensor and Sensor Fusion Technology in Autonomous Vehicles: A Review". In: *Sensors* 21.6 (2021). ISSN: 1424-8220. DOI: `10.3390/s21062140`. URL: `https://www.mdpi.com/1424-8220/21/6/2140`.

[289] Zheqi Yu, Amir M. Abdulghani, Adnan Zahid, et al. "An Overview of Neuromorphic Computing for Artificial Intelligence Enabled Hardware-Based Hopfield Neural Network". In: *IEEE Access* 8 (2020), pp. 67085–67099. DOI: `10.1109/ACCESS.2020.2985839`.

[290] Yunshuang Yuan, Hao Cheng, and Monika Sester. "Keypoints-Based Deep Feature Fusion for Cooperative Vehicle Detection of Autonomous Driving". In: *IEEE Robotics and Automation Letters* 7.2 (2022), pp. 3054–3061. DOI: `10.1109/LRA.2022.3143299`.

[291] Zhou Yuqing, Naoya Niwa, and Hideharu Amano. "Distance Aware Compression for Low Latency High Bandwidth Interconnection Network". In: *2022 IEEE 15th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoC)*. 2022, pp. 361–367. DOI: `10.1109/MCSoC57363.2022.00063`.

[292] Sohaib Zafar. “Unleashing the Power of YOLO v8: A Breakdown of its Working Principle and Evolution”. In: *Medium* (2023). Accessed: 02.01.2024. URL: https://medium.com/@sohaib.zafar522/unleashing-the-power-of-yolo-v8-a-breakdown-of-its-working-principle-and-evolution-14a98d1feedc.

[293] Francesco Zanetto, Fabio Toso, Vittorio Grimaldi, et al. “Time-Multiplexed Control of Programmable Silicon Photonic Circuits Enabled by Monolithic CMOS Electronics”. In: *Laser & Photonics Reviews* 17.11 (Sept. 2023). ISSN: 1863-8899. DOI: 10.1002/lpor.202300124. URL: http://dx.doi.org/10.1002/lpor.202300124.

[294] Sanaz Zarei and Amin Khavasi. “Realization of optical logic gates using on-chip diffractive optical neural networks”. In: *Scientific Reports* 12.1 (2022), p. 15747.

[295] Aston Zhang, Zachary C. Lipton, Mu Li, et al. *Dive into Deep Learning*. https://D2L.ai. Cambridge University Press, 2023.

[296] Feihu Zhang, Daniel Clarke, and Alois Knoll. “Vehicle detection based on LiDAR and camera fusion”. In: *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. 2014, pp. 1620–1625. DOI: 10.1109/ITSC.2014.6957925.

[297] Shifeng Zhang, Longyin Wen, Xiao Bian, et al. “Single-Shot Refinement Neural Network for Object Detection”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2018.

[298] Shiwen Zhao, Wei Wang, Junhui Hou, et al. *Hybrid Pooling and Convolutional Network for Improving Accuracy and Training Convergence Speed in Object Detection*. 2024. arXiv: 2401.01134 [cs.CV].

[299] Zhong-Qiu Zhao, Peng Zheng, Shou-Tao Xu, et al. “Object Detection With Deep Learning: A Review”. In: *IEEE Transactions on Neural Networks and Learning Systems* 30.11 (2019), pp. 3212–3232. DOI: 10.1109/TNNLS.2018.2876865.

[300] S. N. Zheng, J. Zou, H. Cai, et al. “Microring resonator-assisted Fourier transform spectrometer with enhanced resolution and large bandwidth in single chip solution”. In: *Nature Communications* 10.1 (May 2019). ISSN: 2041-1723. DOI: 10.1038/s41467-019-10282-1. URL: http://dx.doi.org/10.1038/s41467-019-10282-1.

[301] Chao Zhou, Yanan Zhang, Jiaxin Chen, et al. “OcTr: Octree-based Transformer for 3D Object Detection”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 5166–5175.

[302] Hailong Zhou, Jianji Dong, Junwei Cheng, et al. “Photonic matrix multiplication lights up photonic accelerator and beyond”. In: *Light: Science & Applications* 11.1 (2022), p. 30.

[303] Yin Zhou and Oncel Tuzel. *VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection*. 2017. arXiv: 1711.06396 [cs.CV].

Appendix

Power Consumption And Sizing Of Photonic MZI Accelerators

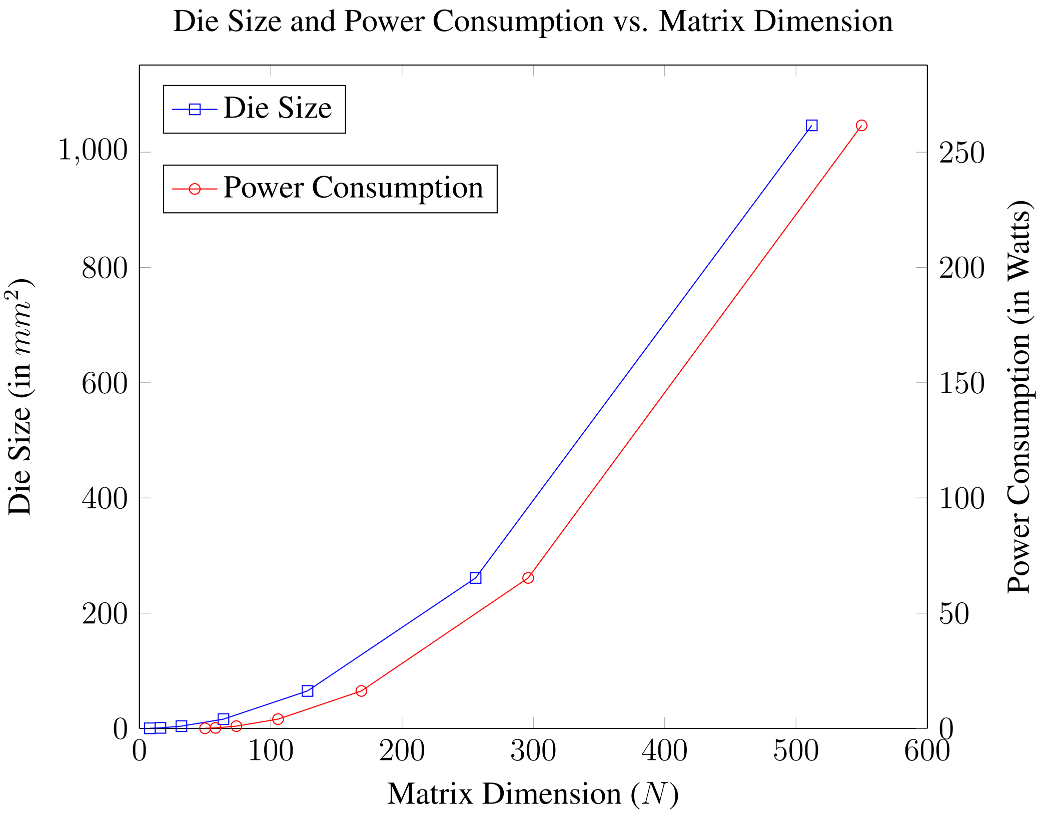


Figure 18: Power consumption and sizing of photonic MZI accelerators

Impact Of Clock-Frequency On OPU Performance

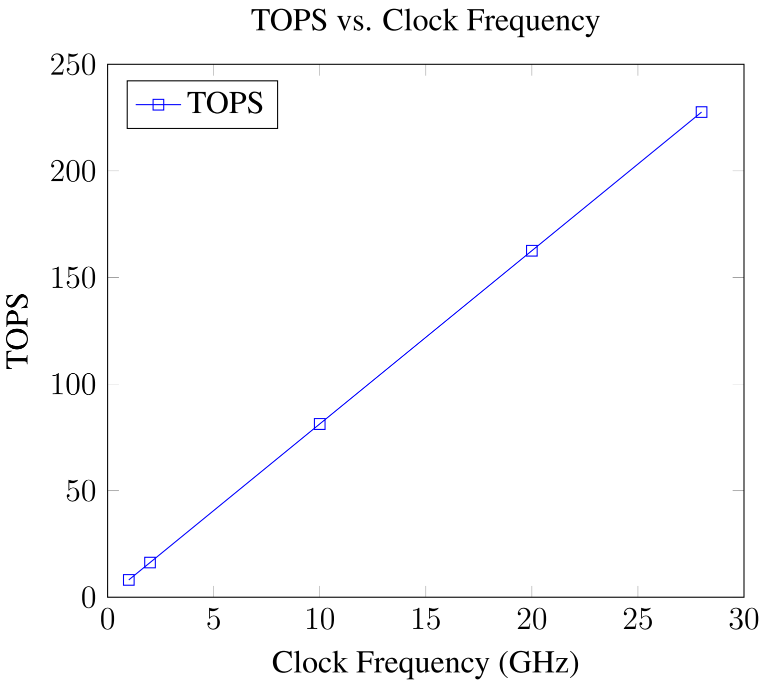


Figure 19: TOPS against clock-frequencies for OPU

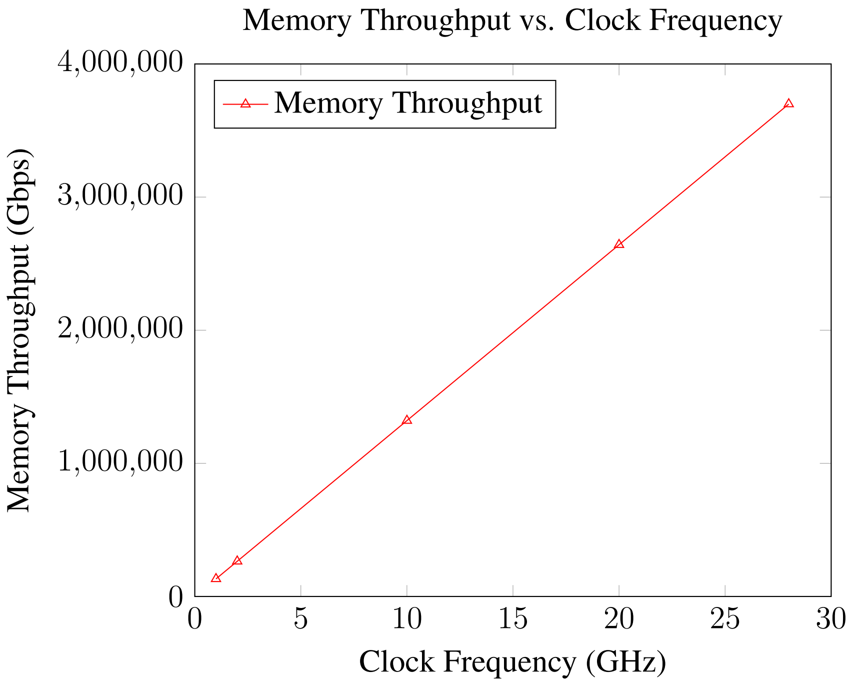


Figure 20: Impact of clock frequency on memory utilization

OSTBAYERISCHE
TECHNISCHE HOCHSCHULE
REGENSBURG

ERKLÄRUNG
ZUR BACHELORARBEIT VON

Name:	Schulenberg
Vorname:	Florian
Studiengang:	Informatik (B.Sc.)

1. Mir ist bekannt, dass dieses Exemplar der Bachelorarbeit als Prüfungsleistung in das Eigentum der Ostbayerischen Technischen Hochschule Regensburg übergeht.

2. Ich erkläre hiermit, dass ich diese Bachelorarbeit selbständig verfasst, noch nicht anderweitig für Prüfungszwecke vorgelegt, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie wörtliche und sinngemäße Zitate als solche gekennzeichnet habe.

Regensburg, den 22.02.2024

F Schulenberg
....

Unterschrift