TECHNICAL WHITEPAPER

Teaching Machines to Detect and Understand Humans

NVISO
ADVANCING HUMAN POTENTIAL

Introduction

The age of autonomous machines and intelligent IoT devices is upon us. They will affect many aspects of our lives. They will bring about a new generation of diagnostic instruments to clinics, improve cars and transportation, and create new consumer experiences, while being embedded into the networks that power and inform society underpinning the efficient provision of public and private services. Increasingly they sense their surroundings, operate autonomously and collaborate with other devices seamlessly.

Critical to their operation, will be improving the safety, security, and quality of life of humans that interact with such intelligent systems. To interact efficiently and effectively, sensing and interpreting human behaviour will become mission critical to connecting the human user with automated systems. This white paper outlines the core horizontal topics necessary to teaching machines to sense and understand humans and use cases where "AI at the Extreme Edge" is not only critical but transformative.

Topics covered in this technical whitepaper include:

- What is **Human Behaviour AI** and what are key **Use Cases**?
- What are **Data Centric AI Apps** and how are they developed?
- What is **Extreme Edge AI Computing**?
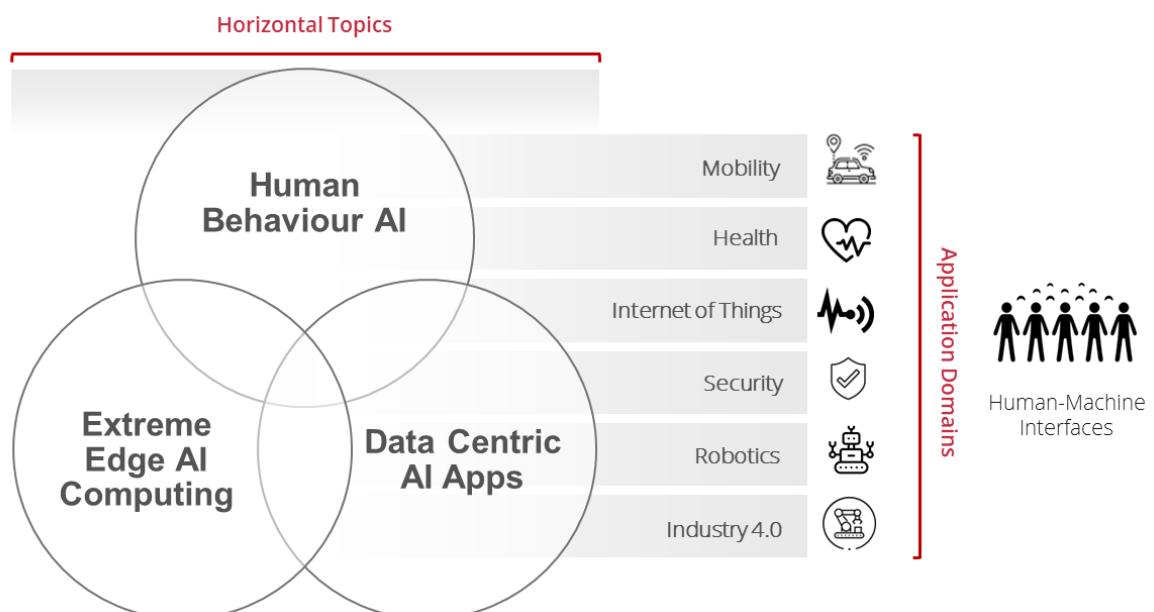
Human Behaviour AI

The task of understanding humans by using smart sensing has become increasingly important in modern society, given the proliferation and ubiquity of embedded sensors in a huge range of devices ranging from smartphones and smart city infrastructure to consumer IoT devices and health monitoring systems . A key to enabling the human user to connect with automated systems is artificial intelligence, which enables the plethora of sensing data to be processed in a way that gives context and meaningful insights to aid smart devices to make decisions and perform tasks.

Although there are a number of challenges in developing and integrating artificial intelligence into smart sensing and IoT applications (ranging from memory footprint and computational complexity, to privacy and robustness), the detection and understanding of human activities using artificial intelligence can be divided into three fundamental layers:

- **Sensors,** which are devices that detect and respond to changes in an environment in which humans are present. Inputs can come from a variety of sources such as light, temperature, motion and pressure (e.g. cameras, microphones, IMUs, etc) which generate data-streams that are used by data-driven and learning-based frameworks.
- **Detectors and Observations**, provide reusable core signals (e.g. finding and locating a face or head in an image) around well understood human modalities directly from sensor data using machine learning models which are not application or industry specific.
- **Reasoning and Semantics**, that apply rules or prior knowledge (nodding of a head may indicate drowsiness or it can be a signal for confirmation) around simple or more complex human states or behaviours that may be context or application specific.
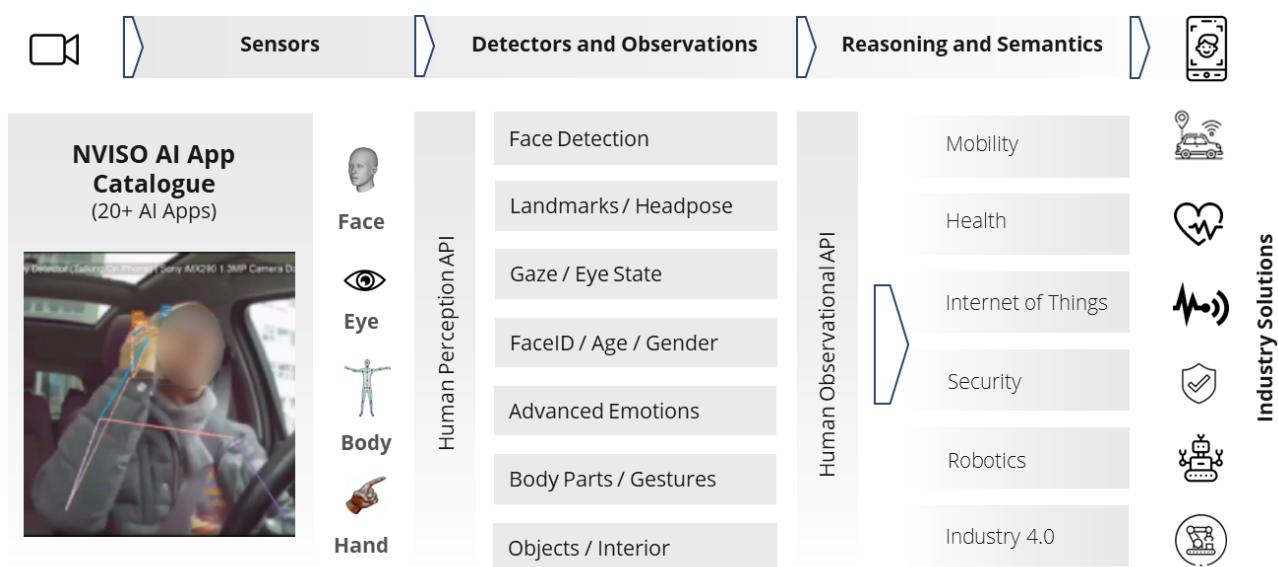
Human Behaviour AI from Sensor to Industry Solution

Detectors and Observations

Non-verbal "expressions" can communicate emotions faster, more subtly and more effectively than words ever can, which is why understanding non-verbal cues remains crucial for systems doing human behavioural analysis. Human detectors and observations analyze among other signals at least the following 4 key areas defined by the face, eyes, body, and hands as follows:

Face

The face is widely regarded as a unique feature of human beings. Humans locate, identify, and distinguish between faces with very casual inspection and this outstanding human ability to identify individual human faces has long been of major interest to cognitive scientists, neuropsychologists, and neuroscientists. The recent advent of smaller, affordable and more powerful computing devices and larger and more diverse facial databases, along with the recent advancements in deep learning have enabled machines to emulate this human capability of identifying and distinguishing the human face often surpassing human level accuracy. Some of the most natural use cases where a machine can make better informed decisions based on the human face analysis are:

- *Identity*: The human face is the significant characteristic for identifying a person. Everyone has their own unique face, even in the case of twins. A face recognition system can find a person's identity through a biometric method.

- *Expressions***:** Human faces can express things that are difficult to put into words. Expressions can communicate emotions faster, more subtly and more effectively than words, which is why understanding facial expressions remains crucial for systems doing human behavioural analysis.

- *Movement*: Head movement (and associated head gestures) are another important part of non-verbal communication in human interaction. Classifications of head movement are based on action type, frequency, amplitude, continuity and other factors.

Face		The core observations that can be modelled and detected from the human faces are:
Core Observations	Location	Facial landmarks are used to localise and represent salient regions of the face, such as: eyes, eyebrows, nose, mouth, jawline.
	Position and Headpose	The rotation and translation vector for the face provides a precise head location and headpose.
	Expression	Universal expressions of emotion which include Anger, Disgust, Fear, Happiness, Neutral, Sadness and Surprise, and Contempt.
	Identity	The estimation for the gender and age along with a unique encoded sequence describing the identity for the face.

Eyes

Human eyes are one of the most important receptors of stimuli allowing humans to perceive their surroundings. In any social interaction involving humans, eyes can play a major role in establishing who we are, how we feel and what we do. Therefore, the ability to perceive and analyse the eyes is a very important capability to have while interacting with humans. In addition, the direction in which eyes are pointed (often referred as Gaze) also provides very significant clues about human behaviour such as attentiveness, distraction, interest and suspicious behaviour. Some of the natural use cases for eye analysis in human-machine interactions are:

- *Identity*: High resolution images of the eyes can be analysed to detect and match the unique iris patterns providing a mode of identification.

- *Attention/Interest*: The eye gaze for humans often tells the point of attention for their visual stimuli. This gaze information combined with other technologies such as object detection can be used to deduce the attention/interest level of human subjects.

- *Cognitive Load*: Often an increase in cognitive load is related to voluntary eye movement such as fixations as well as involuntary eye movements like blinking and pupil dilation. Eye tracking and analysis can be used to detect such behaviours.

Eye 👁		The core observations that can be modelled and detected from the human eyes are:
Core Observations	Eye State	The state of the eyes visible or non-visible including if the eyes are open, closed, or partially open.
	Openness	The level of the openness of the eyes which is measured as a numeric score.
	Blink Rate	The number of blinks per minute.
	Eyelid Closure	The percentage of time during which the eye is closed for a given unit of time. It is one of the most common indicators for fatigue monitoring.
	Gaze	The direction in which eyes are pointed.
	Iris Dilation	The measurement for the dilation of iris in the pupil.

Body

A machine can better interact and help users if it can understand human body poses, actions and activities of people. For example, a robot can take timely actions when it detects the pose of a person who is prone to fall. Therefore, it is very crucial to model and teach computers to understand and interpret the human body pose with a high level of accuracy. With the latest developments on collecting accurate human body pose datasets along with the sophisticated deep learning architectures, it is now possible for machines to accurately detect and analyse human body pose. Some of the natural use cases for body analysis in human-machine interaction are:

- *Joint Location and Shape*: The locations of the body joints for each key point are represented relative to input data in 2D space (i.e., from an image or video frame) so as to understand body location and shape.

- *Position and Movement:* An intuitive human body model that includes a set of 3D joint positions and limb orientations to represent the human body structure and the relations between different body parts for biomechanical analysis.

- *Body Posture:* Classification of static human body postures (sitting, lying, etc).

- *Body Activity Recognition:* Classification of dynamic human activities related to body movements (jumping, walking, standing up, sitting down etc).

- *Volumetric and Weight Analysis*: With the collection of rich datasets having the body volume, weight and pose information, along with advanced deep learning architectures it is possible to detect the salient body features which highly correlate with the body weight and volume. This can be used to adjust machines to provide more customised support to its users.

| Body		The core observations that can be modelled and detected from the human body are:
	Location	Key points on the body which can be used to encode the visible body locations.
	Position and Angles	Set of body joint positions and angles which are needed to perform a specific pose.
Core Observations	Posture	Classification of static human body postures (sitting, lying, etc).
	Activity	Classification of dynamic human activities (jumping, walking, etc).
	Volumetric and Weight	Estimation of body weight and volume.

Hands

Gesturing is a natural and intuitive way to interact with people and the environment. So it makes perfect sense to use also hand gestures as a method of human-machine interaction. Snap your fingers and make your coffee maker brew you a fresh cup. Wave a hand near your smart TV and switch on today's weather forecast. Tap a finger near your smartwatch and set an alarm in your child's bedroom. How great would it be to get things done just by gesturing? It's not that unrealistic anymore: hand tracking and gesture recognition technologies are penetrating multiple industries. Some of the natural use cases for hand analysis in human machine interactions are:

- *Joint Location and Shape*: The "shape" of the hand only considers the locations of the hand and finger joints which are represented relative to input data in 2D space (i.e., image or video frame); it does not consider a hand's orientation in space.

- *Position and Movement:* An intuitive human hand model that includes a set of 3D joint positions and hand and arm orientations used for posture and gesture recognition.

- *Hand Posture:* Classification of static human hand postures (stop, thumbs up, etc).

- *Hand Activity Recognition:* Classification of dynamic human activities related to hand movements (tapping, grasping, clicking, etc).

- *Gesture Recognition:* Classification of dynamic hand and arm movements to infer the underlying gesture performed by the person.

Hand		The core observations that can be modelled and detected from the human hand are:
Core Observations	Location	Key points on the hand which can be used to encode the visible hand locations.
	Position and Angles	Set of hand joint positions and angles which are needed to perform a specific pose.
	Posture	Classification of static human hand postures (stop, thumbs up, etc).
	Gesture	Classification of dynamic hand and arm movements to infer the underlying gesture performed by the person.

Reasoning and Semantics

Human behaviour signals provide important information by themselves useful for machines or computers to detect and understand humans. Additional insights however can be extracted by looking at how the core observations change over time or correlate together and can be used with context and/or application specific reasoning and semantics for industry specific solutions. Core behaviour signals generated from AI Apps (which carry out basic human detections and observations as previously explained) are then combined at an application level with reasoning and semantics to enable improved or new human-machine interactions.
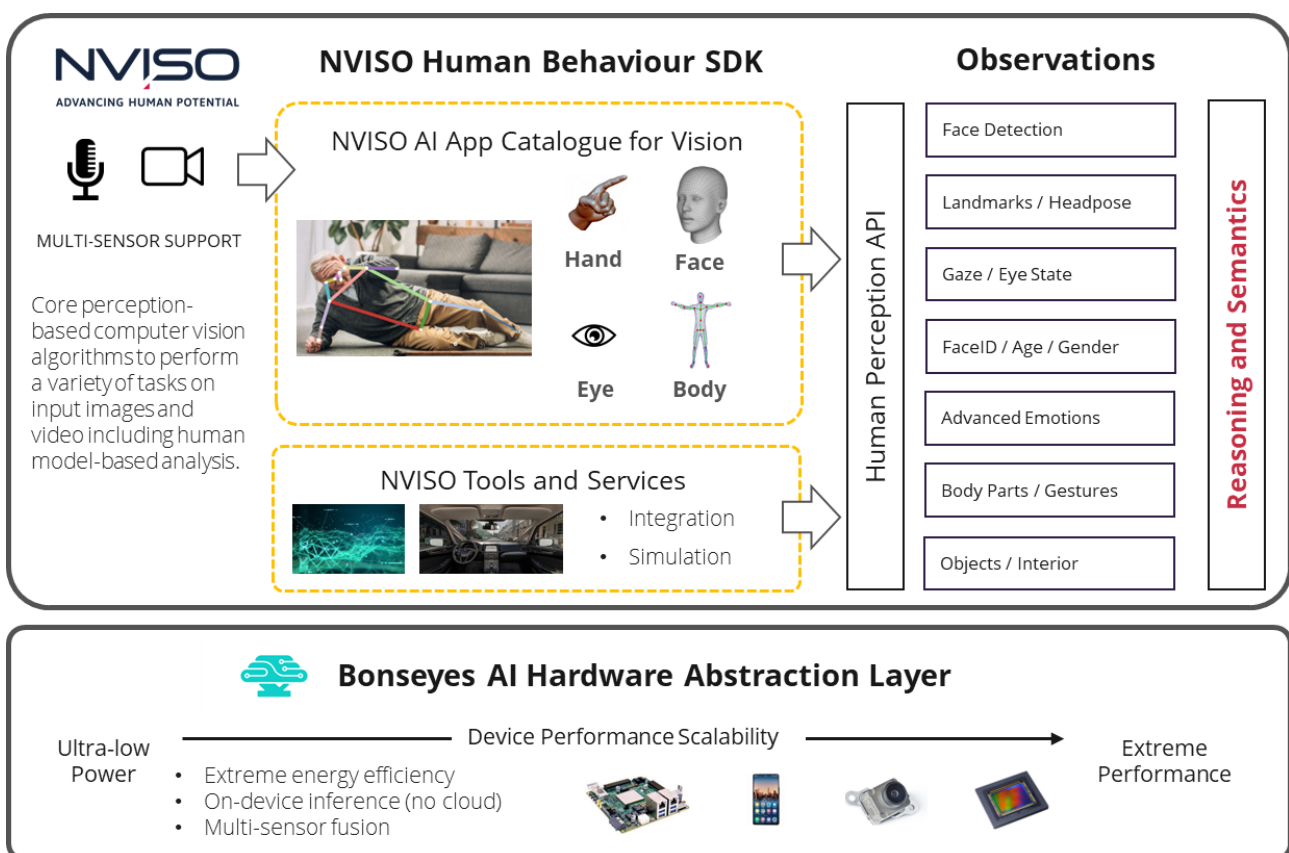
This section outlines some examples (non-exhaustive) of how detections and observations can be combined with reasoning and semantics to create extra layers of interpretation and application value.

- **Activities:** For various reasons, the machines we are building would like to know the activities in which the persons in its field of view are involved. The activities of interest vary widely with the application and the context: from activities which might distract the driver from the road in the automotive context (such as eating, drinking, smoking, speaking, sleeping) to activities any person is involved in when close to a robot's viewpoint (sitting, standing, talking to somebody, walking etc.). Apart from the obvious use case of providing triggers to real time decision making, identifying activities is also important for filtering various core signals to increase their robustness and relevancy.

- **Gestures:** Vision-based AI opens up a whole new level of human-computer interaction. Forget about touch-based inputs like keyboards or touch screens and welcome vision-based gesture control for interacting with the new generation of intelligent machines, be they cars, drones, social robots, wall displays or industrial machinery. Further, without the need for contact-based sensors, vision-based gesture recognition has unchallenged advantages: longer distance, no wearable or additional sensors needed, environmental independence.

- **Distraction, Inattentiveness, and Drowsiness**: Distraction is often related to an object, a person, an idea or an event that diverts someone's attention from a task. Distraction-detection systems use head pose or gaze information to detect if someone is paying sufficient attention to a task. Drowsiness refers to sleepiness, often in inappropriate situations. Although the state of drowsiness may only last for a few minutes, its consequences can be disastrous. Someone does not become drowsy suddenly, without showing some signs. Examples of such signs include difficulty keeping eyes open, yawning, frequent blinking, difficulty concentrating, and nodding. These signs gradually become more apparent as drowsiness deepens and, as such, can serve as indicators for the level of drowsiness.

- **Incapacitation**: Incapacitation can be caused as a result of physical or mental exertion or a prolonged period of performing the same task. The eye states (open, closed, partially closed) combined with the technologies such as expressions and head movement can be used to decipher the level of inability for performing tasks such as driving etc. which can be very beneficial for keeping safety and security at the forefront.

- **Fatigue**: Fatigue is defined as a global reduction in physical or mental arousal that results in a performance deficit and a reduced capacity of performing a task. When a person enters a state of fatigue, two things happen: blinking will unconsciously increase for protection, or the eyes will become sluggish and blurry and the number of blinks will change. The increase or decrease in the number of blinks compared to the state of consciousness can reflect the degree of fatigue.

- **Stress**: Usually there is a visible change in the expressions and eye state of humans upon being exposed to increasing levels of stress. The eye blinking frequency increases as well as the presence of the main facial features related to anger and fear. Combining these two measurements can provide the level of stress.

- **Dangerous and Suspicious Behaviours**: Based on the context, certain behaviours can be classified as dangerous ((like taking the gaze away from the road while driving) or suspicious (like a vehicle cruising the streets repeatedly). By combining the core signals like object detection, body and face analysis it is possible to accurately identify the surroundings and flag the behaviours and activities that are deemed dangerous or suspicious for those environments.

- **Advanced Emotions:** The trigger for a transient emotional response may be identified. Is there another person or object which makes us smile? The head orientation or gaze vector when an emotional response is registered will, in most cases, indicate the direction of stimulus. If another person or object is detected in the direction of attention when the emotional response is registered, it can be postulated that that is the source of the emotional response.

Use Cases

In this section, a number of applications involving the adoption of Human Behaviour on Extreme Edge AI Computing are illustrated and analysed with the goal to highlight the considerable breadth of scenarios where NVISO technology can play an important role.

Smart Health

Data will play an increasingly important role in providing a better understanding of consumer needs in terms of health, and to enhance and tailor a more cost-efficient health offering that delivers the right care at the right time and in the right place. The interconnections made possible by being able to access pools of data not previously available (worldwide databases, data clouds, apps, in-sensor computing etc) are creating a major shift in healthcare provision.

According to different projections, healthcare budgets around the world are expected to increase by 10% in aggregate by 2030. Healthcare spending will be driven by ageing and growing populations, rising labour costs, and by clinical and technology advances. Consequently, by 2030 healthcare is expected to be centred on patients being empowered to prevent disease rather than seek treatment.

Healthcare budgets in Europe will therefore shift towards novel areas such as digital health and more advanced prevention and rehabilitation options, for which homecare will play a key role. Money is expected to be redirected toward personalized medicine for the most complex diseases, and preventive, early-stage treatments. This split is likely to lead to significant changes and require new R&D strategies for many industry players.

- **Vital sign monitoring - detect health problems on the spot:**
  Non-invasive observational sensors (for heart rated, breathing frequency, movement patterns, etc.) are important components in many medical applications. However, a cloud-based implementation of the sensing would be too slow in time critical contexts. This is not the only problem of cloud systems as storing generated data in them is also a privacy concern. Issues of latency and privacy can be solved by using edge AI.
- **Personalised medicine - personal health monitoring anywhere and anytime:**
  Human physiology can vary greatly from individual to individual. Examples for that include blood pressure or lung capacity. However, these differences need to be considered for accurate and robust medical applications like vital sign monitoring. Due to privacy concerns, it is difficult to process this information in cloud-based solutions. Edge AI offers the possibility of maintaining privacy when processing medical data. Furthermore, many medical applications require real time processing, which can be better realized with local AI.
- **Sports analytics - Injury prevention and better performance for athletes:**
  Modern methods to analyse lower-limb injuries are essential for athletes, they are accounting for 77% of hospitalized sport-related injuries and are a risk factor for early-onset osteoarthritis. High-impact forces are one of the factors contributing to lower-limb injuries. To decrease the prevalence of lower-limb injuries, and their associated long-term disability and economic burden, multiple injury prevention programs have been proposed. These consider the study of ground reaction forces (GRFs) to enhance athletes' performance, determine injury-related factors, and evaluate rehabilitation programs' outcomes.

Smart Living

Rethinking human activities to take advantage of the innovation opportunities offered by hyper-connectivity and AI solutions and new kinds of sensors based on miniaturized technologies will create numerous benefits for every new market, ranging from connected cars and digital health to smart home and smart living, and factories of the future. This should include lessons learned from the COVID-19 pandemic like the sudden increase in remote-working. Smart Living is a solution that aims to make an environment of the future that improves people's quality of life and examples include:

- **Companion robots - Personal assistant for everybody:**
  Such robots are designed to assist their owners in their daily lives. Detect presence and identity through facial recognition. Anticipate and react to owner needs by observing head and eye movements, by tracking head pose and gaze. Pay attention to and appropriately adjust to mood through observation of the owner's emotional state. Especially in Japan it is a very popular concept addressing the needs of lonely, elderly people, a market segment which is expected grow on a global scale.
- **Metaverse - a personalized virtual world:**
  Augmented reality (AR) and virtual reality (VR) along with artificial intelligence and blockchains are a key element to create this virtual world. One of the most interesting and spoken about concepts of the Metaverse is Avatars. People are creative and love the thought of designing themselves in a virtual world. They can change their hair colour, style of clothing to their preference. AI could analyse 2D user images or 3D scans to create very realistic and accurate Avatars. Companies such as Ready Player Me have already actively been using AI to help build Avatars for the Metaverse.
- **Affective computing - Body language and emotions are key to understand humans:**
  Many applications rely on automatically detecting and recognizing the emotional state of a human, either with remote or "nearable" sensors (visible and IR imagery, audio, physiology), or with sensors in contact (wearables) for physiology, or activity monitoring. Emotions, a classic conceptual representation of which follows a 2D valence (negative / positive) versus intensity (calm / excited) pattern, have an essential role in human behaviour. These influence the mechanisms of perception, attention, decision making, and social behaviour. The purpose of estimating emotional states is to improve understanding of human behaviour. This is the strongest reason as emotional states are both very personal and evolving, very different from one individual to another, and from one situation to another. Edge AI allows for maintaining the confidentiality of the data inside the measurement device, to guarantee the autonomy of the devices, and to aim for an individual estimator learning over time.

Smart Mobility

Mobility is a basic human need and Europe's mobility industry is a key contributor to it. The usage of smart perception, safety and automated mobility solutions and services to provide safe and comfortable inclusive mobility that is also suitable for the elderly as well as people with special needs. Research, development, and innovation (R&D&I) of embedded AI-based software, sensors and electronic components and systems provide the core of automated on- and off-road vehicles, ships, trains and airplanes. A special focus requires validation of the safety and reliability of the automated

mobility systems in all traffic and environmental situations as there are currently no adequate methods and tools available.

- **Occupant monitoring - More safety and comfort in cars and public transportation:**
  This is expected to become a key feature and a standard in new cars as a result of regulatory and rating agency requirements Euro NCAP 2025 Roadmap. Advanced driver monitoring systems (DMS) can detect distracted and drowsy drivers by accurately measuring eye and head position, attention, and fatigue. The DMS alerts the driver and integrated safety systems upon detection of a risk such as drowsiness or distraction. This feedback enables the driver and vehicle to act before safety is compromised. Managing transitions between different levels of autonomy is fundamental. The AI-based observer is a key point of this system as it detects the behaviour and the mental state of drivers and passengers. Edge AI offers the local calculation of driver and passenger states, thus allowing for control of the response and transition times, while at the same time preventing personal data from leaving the vehicle and continuous learning to adapt the AI-based observer to each driver and passenger.

Robotics

Industry 4.0 changes to the mode of operation have a profound impact on how the factories, construction zones and processes are managed and operated. Powerful networked digital tools are needed to achieve the necessary Situational Awareness and control of autonomous vehicles, robots and processes at various autonomy levels.

- **Collaborative and assistive robots - An untapped potential in many industries:**
  Specialized collaborative robots' industrial environments support human workforce in the fulfilment of repetitive jobs or heavy lifting, for instance. Applications can be found mainly in the manufacturing industry, e.g., assembly of automotive parts. But next to industrial application there exist several opportunities in care applications as well, e.g., to grip a glass of water. Edge AI enables new possibilities for the cooperation of humans and robots, because in contrast to cloud-based systems edge AI is fast enough to handle situations where the robot could inflict harm. To implement these new possibilities, sensors need to be deployed that can monitor the environment and the movement of humans and animals within range. The data from these sensors are locally processed by AI models in the robot or running on nearby edge nodes. Afterwards, edge AI-based components use the processed data to control the robot, allowing for close cooperation with humans in performing complex tasks like the manufacturing of custom products in workshops or rescue operations. To implement this vision of close cooperation many challenges need to be solved such as training the robots for new tasks.
- **Autonomous robotics - Driving the efficiency of future work forces:**
 This type of robot is advancing rapidly and becoming robust and reliable in a number of fields, but there remains much to be done before intelligent robots can work together with us as assistants. Today, most robots are either tele-operated or perform precisely defined missions. In many cases the human overhead required to use a robot exceeds the usefulness of having it. There is a clear need for human-robot teams in which humans and robots work

side-by-side. Homogeneous teams involve team members doing the same work. In such a case, the value comes from parallelism, such as picking grapes from vines.

Data Centric AI Apps

What are AI Apps?

An Artificial Intelligence Application (AI App) is the entire software including any machine learning component(s) of a data-driven algorithm necessary to transform data input(s) into output(s). An AI App is represented by its architecture that shows how to transform its input(s) into its output(s) where the transformation is given in the form of learning algorithm(s) (normally specified in frameworks such as PyTorch, TensorFlow, etc.) including pre and post processing algorithms and code. Data Centric AI is the discipline of systematically engineering data used to build and continuously improve an AI system over time.
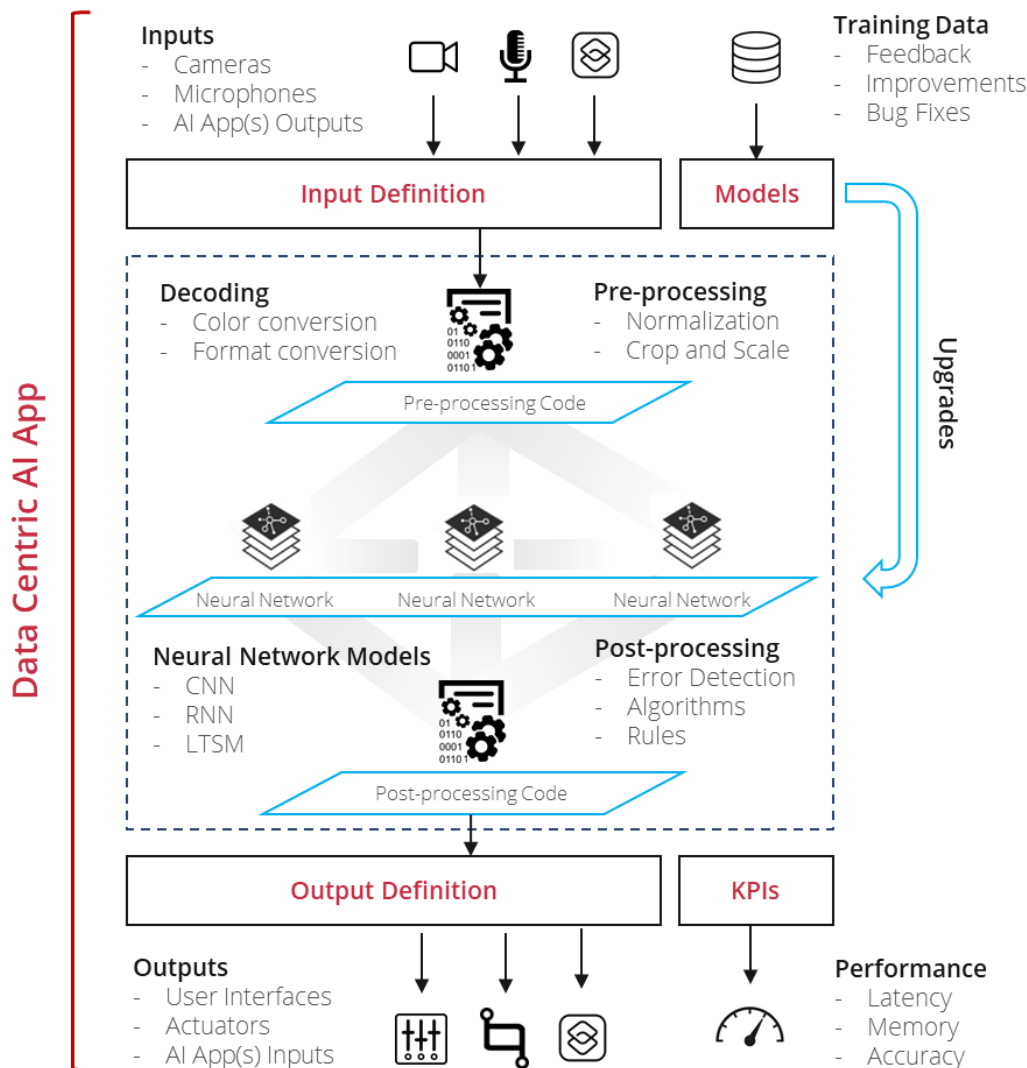
What is the virtuous cycle of AI Apps?

The virtuous cycle of AI Apps, also called the "AI Flywheel Effect," is one of the most exciting ideas in Artificial Intelligence and it's also incredibly simple. Essentially, when AI technologies are integrated with a product properly, they create a feedback loop where the product continuously improves with use, generating more usage and a better competitive position relative to other products.
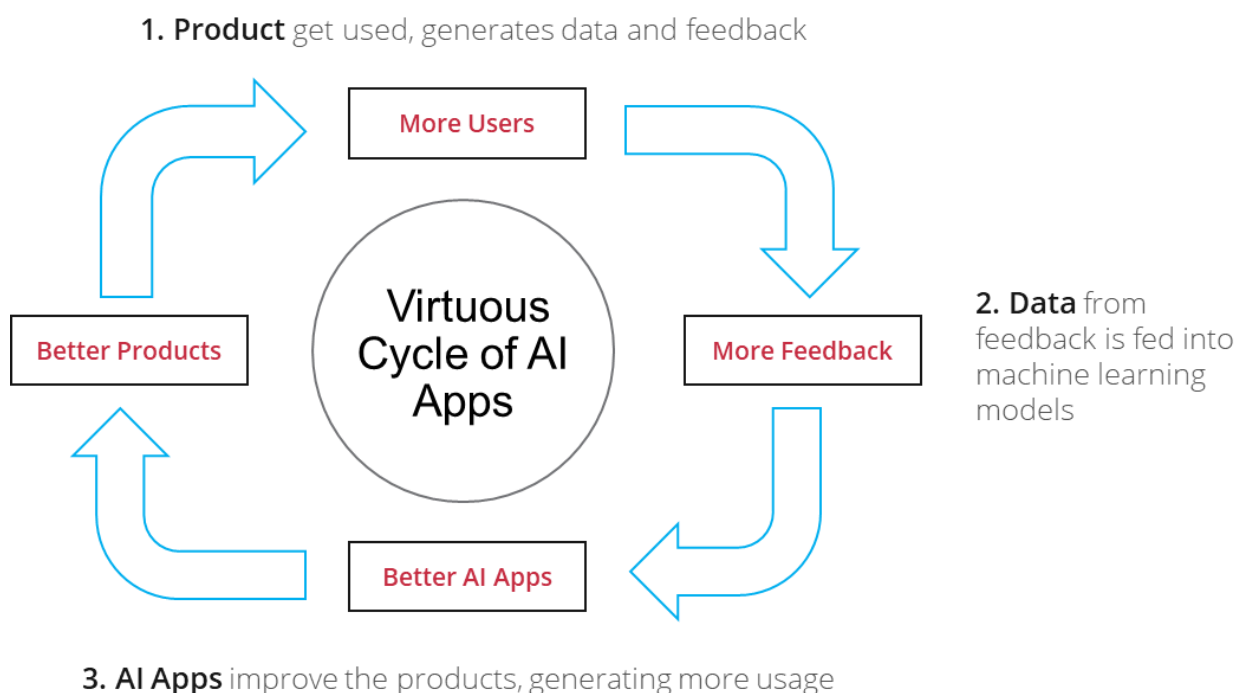
Although any product tends to improve with usage regardless of its underlying technology because a good team will use qualitative feedback and analytics data to bring it closer in line with user needs. This improvement, though, tends to reach an asymptote where additional usage and data no longer provide much marginal insight to the product.
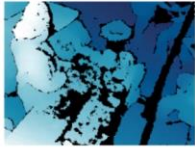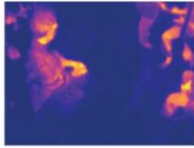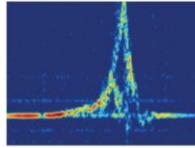
When a product's core technology is AI-driven, though, it adds another layer on top of the typical team-driven product improvement cycle. With today's AI technologies (for example Deep Learning), additional data continues to provide marginal improvement for a very long time, allowing the cycle to continue much further than it previously could and improving the core product for a long time.

What sensor signals can be used as input with AI Apps?

Humans observe the world in real-time through a combination of different modalities (vision, hearing, touch, taste, and smell) and can respond to a range of physical stimuli such as light, sound, gravity and acceleration, chemical substance, position, motion, and temperature. Machines, on the other hand, interpret the world through data that algorithms can be taught to process. To sense the presence of humans, a variety of sensors are used ranging from cameras to detect colour, depth, and temperature information. Radar and Wi-Fi to detect distance and velocity. Microphones to detect sounds and speech. Each sensor type and signal have specific characteristics (see below), and usage varies on use case and application requirements and constraints. However, sensor signals for the detection of humans generally falls into three broad categories:

- Analysis of **visual information** (e.g., camera, depth, thermal). Over the past few years, quality mobile cameras have proliferated in devices ranging from smartphones, surveillance devices, and robotic vehicles, including autonomous and connected cars.

- Analysis of **audio information** (e.g., microphone). Conversation is natural and that's why it is a primary interface for human-machine interaction. Voice-based personal assistants (VPAs) are growing in popularity in smartphones, smart speakers, smart watches, wireless earbuds, cars, smart TVs and their remote controls.

- Analysis of **time-series information** (e.g., position and motion). Inertial Measurement Units (IMUs) within everyday consumer wearables (e.g. smartwatch or phone) offer a convenient and low-cost method to monitor the natural behaviour of people.

- Analysis of **meta-data information** (e.g., AI App outputs). AI Apps themselves can be used as "sensors" and can generate additional data that can be fed into AI Apps for further processing.

Human Behaviour AI can use one or all categories of data from such sensors.

	Camera	Depth sensor	Thermal sensor	Radio sensor	Acoustic sensor
Sensory information	RGB, colour, video	Lidar	Infrared	Radar, Wi-Fi	Microphone
Function	Measures colour (visible light)	Measures distance to objects	Measures surface temperature	Estimates distance and velocity	Measures air pressure waves (sound)
Sampling rate	30 Hz (1,920 × 1,080)	30 Hz (1,280 × 720)	10 Hz (640 × 480)	800 Hz	44.1 kHz
Bit depth	24 bits	16 bits	16 bits	32 bits	16 bits
Uses	Object recognition, person detection	3D object detection, robotic navigation	Night vision, equipment safety	Motion detection, object detection	Speech recognition, event detection
Data visualization					

From: Illuminating the dark spaces of healthcare with ambient intelligence

How are AI Apps developed and tested?

To develop AI Apps for the Edge is far more demanding due to the limitations in terms of computation and storage abilities when compared to cloud-based systems. Additionally, Edge systems have a wide range of hardware and software stacks which demands rigorous evaluation and testing to ensure correct performance with onboard sensors. While optimal model training and inference on resource-scarce devices are still a debated problem throughout academia and industry, there are four main steps in development and testing of AI Apps (although the actual process can be a more complicated depending on a which product development life cycle model is employed during development e.g. for medical or automotive products):

- **Datasets, Training, and Optimisation**: Given the importance of data-to-data centric AI, it is crucial to ensure the quality and quantity of data that is used as the starting point of any AI App development. Once a training dataset(s) are ready, machine learning algorithm(s) are applied to make the neural network model understand the relationship between the data and the target objective. For edge-based systems the trained models are further compressed and optimised for edge constraints such as memory, latency, and power efficiency.
- **Deployment**: Deployment is the step of porting the AI App from the training environment to the target system, e.g., a low-power microcontroller, using code generation and cross-compilation.
- **Testing and Benchmarking:** The process of testing the trained and optimised model to make sure that it has been well trained and can generalise to unseen data on a target software and hardware platform with software-in-the-loop (SIL) and hardware-in-the-loop (HIL) testing. Verification on the target system of the expected performance is obtained with onboard sensors.

Datasets, Training, and Optimisation
Collect/Annotate/Train/Optimise

Deployment
Code/Compile

Testing and Benchmarking
SIL/HIL Tests and Benchmark

1

Data
Collection and labeling of data. Data augmentation and export of training and benchmarking datasets.

Training
Apply algorithm on training data.

Optimization
Compress and optimize models.

2

Target Device
Compile code generated for target hardware

3

Testing and Benchmarking
Software and hardware-in-the-loop testing with KPI generation

Sensitivity and Bias Analysis
Testing and analysis for bias and sensitivity to noise.

Report : Feedback for next KPI target.

Datasets, Training, and Optimization

Data is the fundamental part of any AI App development workflow. Data refers to representative examples or cases from the target domain that characterises the problem to be solved. In AI modelling, a dataset is composed of many samples that are fed to an AI model, which will infer or predict a target element, e.g., classification, regression, etc. A dataset is an unstructured collection of data generally associated with a set of problems to solve. Such a dataset needs to be structured by means of a Datatool, which provides meaningful and reusable structured samples to train an AI model for such a problem.

Despite having such importance, machines do not understand what data represents. They do not understand why 'a' is 'a' and why it is written in this way or why 'this' means what it means. Most of us do not understand the food that we eat. The only thing that we know is that we have to eat, and we do so. Data for machine learning is food. The machine just consumes it and then learns the relations between different data to predict something rather than understanding the data itself.

Given the importance of data to AI modelling, it is crucial to ensure the quality and quantity of data just as we humans ensure this for our food intake. The process of aiming to prepare and provide the required data to the modelling process is often termed as "Data Engineering" with different steps of this process often together called a "Data Pipeline". Some of the different steps in a typical data pipeline are as follows:

- **Data Identification:** is the process of identifying the right data for solving a specific problem. For example, to build an object detector, it is necessary to have an image-based dataset with images having one or more objects inside them.
- **Data Acquisition and Annotation:** is the process of acquiring the required data once data identification has been completed and adding labels to the data elements by reviewing the elements with the help of problem specific domain experts and tools.
- **Data Harmonisation:** Often due to a wide variety of data collection sources and protocols, the collected datasets have substantial differences in their structure, which makes it difficult to use them in model training. Data harmonisation, by means of a Datatool, is the process of transforming and structuring the collected dataset so that it is easier to analyse and consume by the subsequent model training pipeline.
- **Data Validation and Verification:** Once data has been annotated and transformed, it is very important to ensure the correctness and quality of the dataset. The process of data validation and verification applies multiple methods and approaches to ensure that data used for model training is both correct and accurate, as incorrect, or inaccurate data is of little use.
- **Data Provisioning and Versioning:** is the process of making the dataset available for model building and evaluation such that the data from the dataset can easily be integrated into model training pipelines.

Training is the process of applying the machine learning algorithm on training data to make the AI model(s) understand the relationship between the data and the target objective. In a nutshell, during model training data elements are fed to the AI model iteratively in a random order while making small adjustments to the AI model itself. This iterative process is called "Model fitting", which generally

lasts until the model learns to correctly predict the right outcome for the vast majority of the training data elements. The different steps in the model training are:

- **Data Preparation:** is the process of preparing the dataset's before they are fed to the AI model. This includes partitioning the dataset into 2 or more subsets, often called "Training, Validation and Test" sets. Another major step in data preparation is data balancing which is essentially the process for ensuring that one segment of the data population is not overrepresented compared to any other segment. Overall data preparation is crucial to achieving a trained model that generalises well when applied to unseen data examples.
- **Algorithm Selection:** is the process of identifying the right model architecture(s) and algorithm(s) to solve a specific problem. Selection of the right model architecture(s) depends on the specific problem statement as well as the associated data to be used for the model training. If not selected correctly, the wrong architecture can severely affect the model performance when applied on unseen data examples.
- **Model Fitting:** is the iterative process of feeding the data examples to the AI model in a random order while making small adjustments to the model itself based on its current prediction accuracy. The process generally lasts until the model learns to correctly predict the outcome for most of the data examples.
- **Model Selection:** In general, when performing model training, there are many parameters which can be altered and whose value can influence the final performance of the model. The set of such parameters is called "Model Hyperparameters". Model selection is the process of trying different sets of hyperparameters to train models and selecting the model having the best performance given the problem. Model selection can also have a great impact in the latency and memory consumption on the target device.
- **Model Evaluation:** Evaluation is the process of testing the model to make sure that it has been well trained and can generalise to unseen data. Model evaluation is performed on data that has not been used during training, either a different split of the same dataset or a different dataset (on the same AI task). The difference between the trained error (training) and the test error (evaluation) tells the fitness of the model. Underfitting or overfitting are some of the problems that can be found through evaluating the model on unseen data.

Trained models for cloud-based systems are usually overparameterized, i.e., they are larger than needed, or exhibit redundancy. When ample system resources are available, this is not an issue, however for Edge based systems it is possible to compress and optimise models without any drop in accuracy to reduce memory, latency, and power consumption. Some of the well-known techniques considered for model compression and optimisation include:

- **Compact Model Design:** The Model architecture has a large influence in the latency and memory consumption during the Model's execution on the device. Hence, searching for an efficient and compact design is very beneficial to reduce the Model's requirements on resource-constrained devices.
- **Knowledge Distillation:** Whereby a small (easy to implement) model (student) is trained to behave like a larger trained neural network (teacher) while trying to preserve the accuracy of the teacher model, thus enabling the deployment of such models on small devices.
- **Compression:** Steps such as quantization, sparsification, dimensionality reduction, tensor decomposition, pruning, components sharing, etc. These methods exploit the inherent

sparsity structure of gradients and weights to reduce the memory and channel occupation as much as possible, while improving the arithmetic intensity and efficiency.

- **Hardware-in-the-Loop Co-Design:** Devices exhibit large differences in terms of hardware architecture, computing processor or memory. Hardware-in-the-loop Co-Design allows the adaptation of the Model to the target environment by feeding back to the design environment metrics such as the latency and memory consumption on the device.

Deployment

Deployment is the step of porting the trained model from the training environment to the target device, e.g., a low-power microcontroller including the implementation of all necessary pre and post-processing steps of the AI App. This is a complex step as it involves several data and code conversions as well as low-level optimisations. Some of the steps during deployment are the following:

- **Device Tooling Setup and Configuration**: Unlike cloud-systems, low-power devices are very heterogeneous and come in many different forms. Setting up a device is a difficult step that requires very detailed and specific knowledge about hardware and software. Thus, easy-to-follow workflows, i.e., Developer Board Packages, containing the full software stack (operating system, drivers, middleware components, etc.) and documentation required to procure, set up and control target hardware are of extremely importance to the users.
- **Code Generation:** Models are usually trained in Python (high-level language) that allows an easy design process. However, Python is not suitable and optimised for low-power microcontrollers. Hence, a code generation step is needed to convert the representation of the model to a low-level language (C-code) that is understandable by the target device. Code generation also includes the implementation of necessary pre and post-processing steps.
- **Data Format Conversion:** Models may be organised in different data formats or layouts, exploiting the inherent structure of the data. Likewise, data can be represented by a different number of bits (Quantisation), trading precision with energy consumption. During deployment, certain design options need to be taken that optimise the deployment of the model on the target device.
- **Memory Allocations:** Memory Allocation process reserves memory space to the input/output tensors and the weights of the neural network. Since memory is very scarce in low-power microcontrollers, an efficient memory allocation strategy is fundamental to allow the deployment of large Models on these resource-constrained devices.

Testing and Benchmarking

Once an AI App is deployed to a target device testing and benchmarking needs to be carried out across the deployment chain to assess the performance of the model after optimisations and deployment on target devices which also includes evaluation of pre- and post-processing steps integrated within the AI App interface. Testing and benchmarking is performance under industry standard Software-in-the-Loop (SIL) and Hardware-in-the-loop (HIL) testing protocols:

- **Software-in-the-Loop (SIL) Testing:** SIL testing assesses that the AI App still performs correctly after porting from the training environment and after the optimisation and code

generation process for embedded devices. As the model(s) inference results may vary as different software libraries are used for model inference, it is therefore needed to perform careful evaluations to make sure no drop in performance occurs due to the complete software environment and inputs.

- **Hardware-in-the-loop (HIL) Testing:** HIL testing is performed to the AI App when it has been deployed on the target hardware environment with inputs from sensors and running on the target environment within a simulation environment. HIL testing entails simulating device and environmental inputs for the hardware under test, causing it to believe that it is reacting to real-world conditions in the real-world. The HIL bench contains all of the relevant device components. A simulator presents inputs to actual cameras and sensor systems, which in turn send signals to the system under test to see whether it responds correctly to the inputs. HIL testing confirmation of the performance at this stage will guarantee the correct functioning of the AI App for the target application meeting memory, latency, and power efficiency requirements.

What is the typical data flow of an AI App?

The training of AI Apps requires training datasets that often require a series of steps to be executed before and after the neural network models which must be performed identically at inference in order to obtain accurate predictions. These data flow steps can be summarized as follows:

- **Decode:** Input data from image sensors must be decoded and converted into a format which is compatible with the AI App. e.g. conversion of RGB input data from videos to a single monochrome channel.
- **Pre-processing:** Model training can be more efficient if certain pre-processing steps are performed on the network inputs. These pre-processing steps must be replicated exactly at execution, as with the same decoding step.
- **Neural Network Model(s):** A set of parameters, known as weights, and mathematical operators described as a neural network graph determine the type of model used in the AI App. Common types include Deep Neural Networks (DNN), Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Long/Short-Term Memory Networks (LSTM).
- **Post-processing:** Additional algorithms and rules can be executed on the model outputs to improve and create additional predictions.
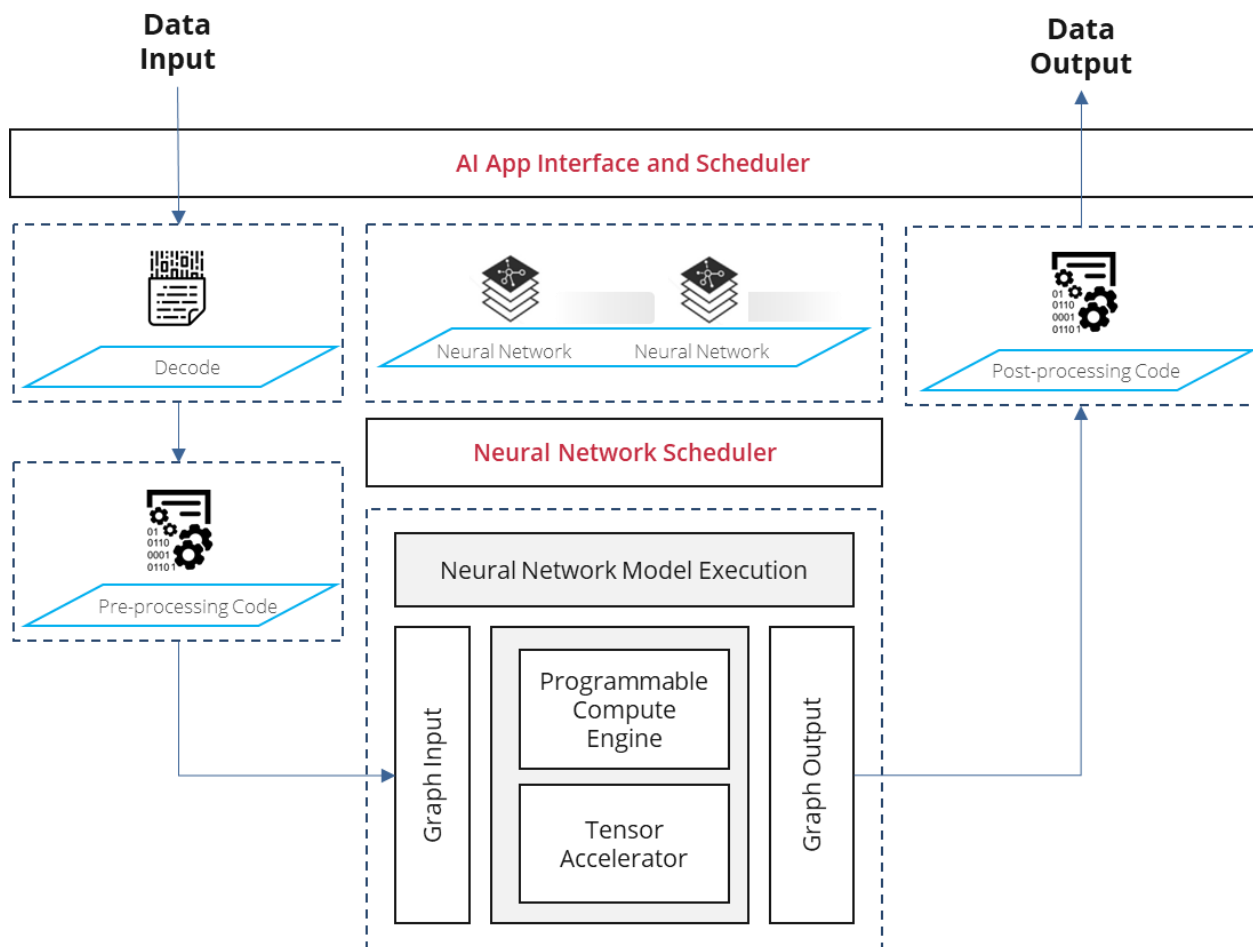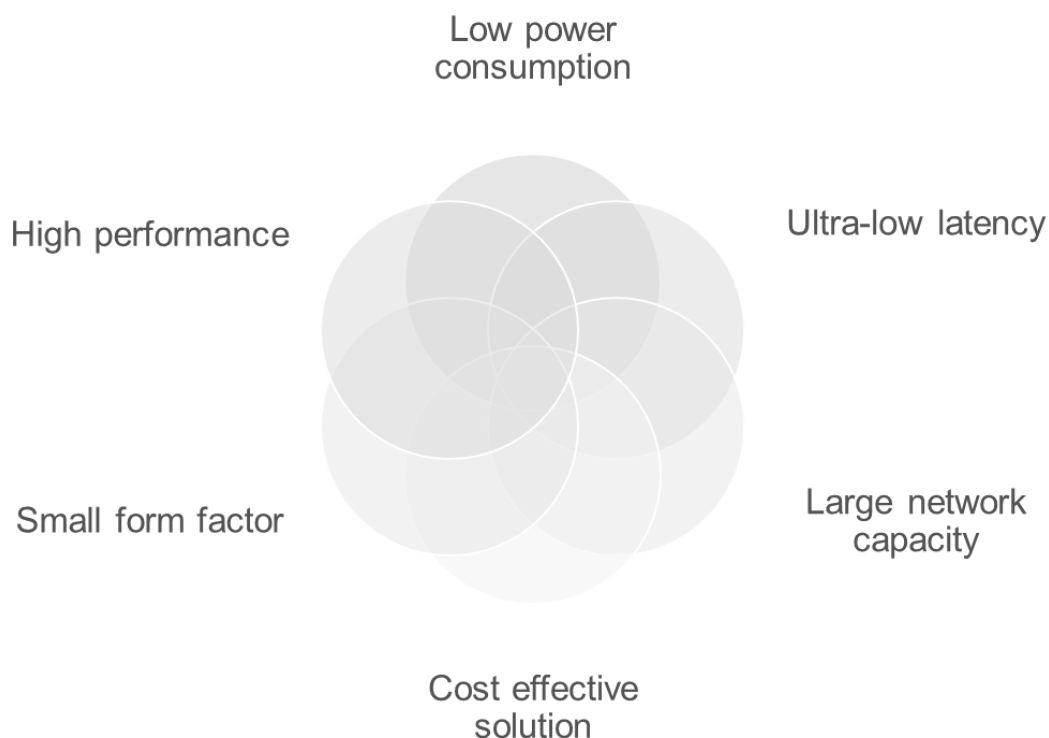
Extreme Edge AI Computing

With the explosive growth in the adoption of Artificial Intelligence (AI) to address a large range of problems that were deemed very hard, data-intensive applications have placed a high demand on hardware performance, in terms of short access latency, high capacity, large bandwidth, low cost, and ability to execute artificial intelligence (AI) tasks. So far, the majority of attention and investment has been directed towards "Cloud AI", with "Data" being the largest common denominator in creating value for industries, governments, and individuals' lives. However, the quest for intelligence is fast becoming a prominent and essential feature at the "Edge" as well, where trillions of "things" will combine to generate even more data. Given the severe constraints that govern edge devices in terms of efficiency, footprint, robustness and cost, it is self-evident that bringing true intelligence to the edge will require profound innovation at all levels of the stack from the computational concepts all the way down the implementation technology.
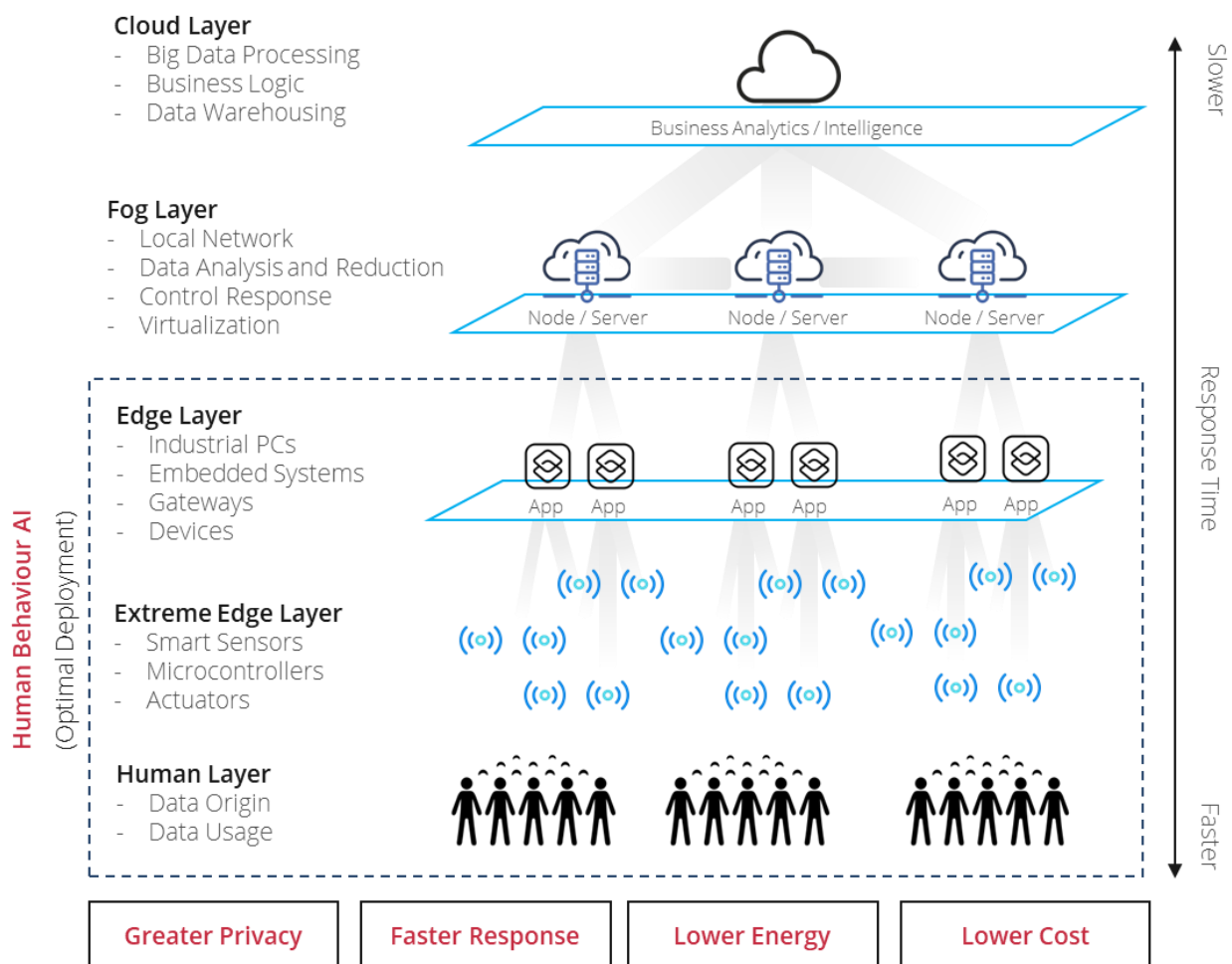
What is "AI at the Extreme Edge"

Edge AI means that AI software algorithms are processed locally on a hardware device. The algorithms are using data (sensor data or signals) that are created on the device. A device using Edge AI software does not need to be connected in order to work properly, it can process data and take decisions independently without an internet connection. Edge AI applications need to be low-cost, small-form-factor devices with low latency, high performance, and low power. In many cases they are even battery operated and mobile resp. portable and need to perform on hardware with limited compute power.

**The Challenges of Meeting
Extreme Edge AI Requirements**

Low power
consumption

High performance

Ultra-low latency

Small form factor

Large network
capacity

Cost effective
solution

Edge computing is emerging as a strong alternative to traditional cloud computing, enabling new types of applications with the advantage of implementing the required AI solutions as close as possible to the end-users and the data sources. Edge devices often face extremely tight constraints in terms of size and energy budget where the results of the data analysis serve as inputs to a controller and are translated into concrete and immediate action within a very tight latency window. This is the case in applications such as autonomous mobile devices (cars, drones), robotics, human-machine interfaces (AR/VR), brain-machine interfaces and wearable medical devices.

While one can easily imagine that parts of such systems can be run from a centralised location, energy, latency, robustness, security, and privacy considerations form powerful arguments for more distributed realisations with a large share of the functionality executed on the Edge (or a

cluster/swarm thereof), and eventually penetrating all the way into the Extreme Edge (sensors and actuators for in-sensor computing).
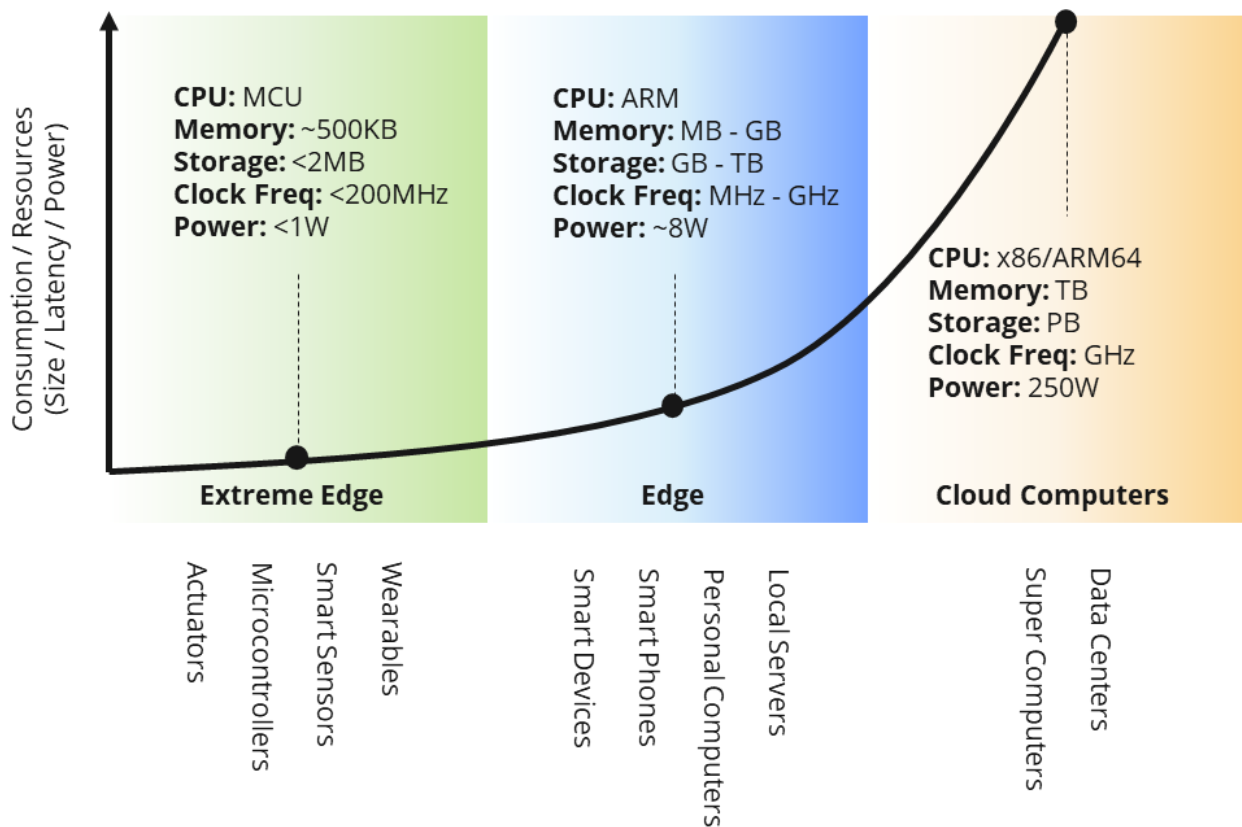
Benefits of "AI at the Extreme Edge"

AI solutions that run autonomously, and are distributed and implemented **at the Extreme Edge** offer the following advantages:

- **Increased real-time performance (low latency):** Edge applications process data and generate results locally on the sensing device. Consequently, the device is not required to be continuously connected with a cloud data centre. As it can process data and make decisions independently, there is increased real-time performance in the decision-making process, reduced delay of data transmissions and improved response speed.
- **Reliable low-bandwidth communication:** Distributed devices can handle many computational tasks, therefore reducing the need to send data to the cloud for storage and further processing. Overall, this results in minimising the traffic load in the network and supports low-bandwidth communication.
- **Enhanced power-efficiency:** As the amount and rate of data exchange with the cloud is minimised, the power consumption of the device is reduced thus improving battery lifetime, which is critical for many edge devices.
- **Improved data security and privacy:** By processing data locally with Edge computing it does not have to be sent over a network to remote servers for processing. This improves

data security and privacy as it can perform all processing disconnected from the central server, which is a more secure and private architecture.

- **High availability**: Decentralisation and offline capabilities make edge AI more robust since internet access is not required for processing data. This results in higher availability and reliability for mission-critical, production-grade AI applications.
- **Lower costs:** Edge AI has analytical scalability and reduced latency, which can lead to significant cost reductions for businesses. Processing and analysing large amounts of data in the cloud is not cheap. For example, cloud companies charge for every minute of inference per endpoint. That can be helpful for organisations that want to pay on demand, but for organisations that require a lot of real time processing (for example, smart cities or hospitals with many cameras and sensors running 24 hours a day), the burden becomes very heavy.

Market Opportunities of "AI at the Extreme Edge"

According to the study published by Astute Analytica, the Global Edge AI Software Market is projected to witness a **major rise in its revenue from US$ 1,459.8 Mn in 2021 to US$ 8,049.8 Mn by 2027**. The market is registering a CAGR of 29.8% over the forecast period. Various factors such as increasing enterprise workloads on the cloud and rapid growth in intelligent applications are expected to drive the adoption of edge AI solutions and services. The Edge AI hardware and the consulting market will grow at the same pace. Grand View Research estimates that the total global Edge computing market will grow 37.4 percent per year and will be worth **$43.4 billion by 2027**.

Source: European Commission, Towards 5G.

Key drivers of extreme-edge AI computing include:

- **Advanced Communication Technologies:** 5G networks can collect large and fast data streams. The construction of 5G networks is beginning gradually, and they will be established

initially in very local and densely populated areas. When the utilisation and analysis of these data streams are as close as possible to the devices connected to the 5G network, the value of Edge AI technology will increase.

- **Massive Amounts of IoT Generated Data:** Currently, about 5%–15% of the world's energy is spent in some form of data manipulation, such as transmission or processing [1], and this fraction is expected to rapidly increase due to the exponential increase of data generated by ubiquitous sensors in the era of internet of things. If the data is separated from its source and there is no metadata describing the meaning of the data, the data will tell us nothing. Therefore, recovering data is not enough. Only Edge AI can take full advantage of advertised IoT data. Large amounts of sensor data can be analysed locally, and operational decisions can be performed automatically. Only the most important data is stored in a data warehouse in the cloud or data centre.

- **Customer Experience:** People expect a smooth and flowing service experience. Today, a delay of just a few seconds can easily ruin the customer experience. Edge computing responds to this demand by eliminating the delay caused by data transmission. Additionally, sensors, cameras, GPU processors and other hardware continue to be cheaper, so more and more people are able to use highly productive and custom edge AI solutions.

Hardware for Edge AI

Choosing the ideal hardware for a particular application requires careful consideration of all the requirements. A successful system design finds a balance between the different aspects of system architecture, such as memory footprint, executing time, model accuracy, power consumption, scalability, cost, and maintainability. While data centers allow engineers to scale available computational power to the current demand (via GPUs or TPUs), an application running on edge devices needs to keep sufficient power reserves. An increasing number of vendors are now moving from producing simple resource-constrained microcontrollers (e.g. ARM Cortex MCU) to pair general-purpose processors with specialised units tailored to execute the computational tasks required to implement AI solutions. As embedded systems are typically focused on using AI in the form of machine learning (ML) for interpreting incoming sensor data, these specialised sub-processors aim to speed up classification or prediction tasks while maintaining a low power draw. This is especially important in applications running on battery power or with a low potential for cooling the system.
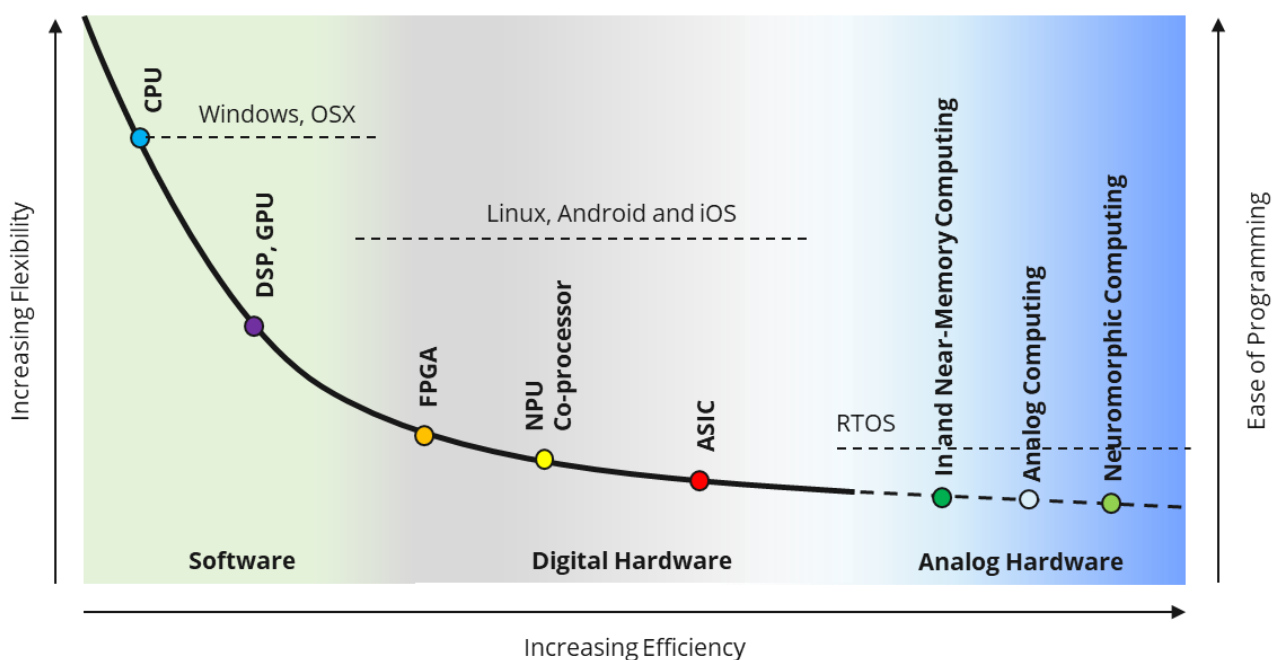
Source: NVISO adaptation of "Computer vision algorithms and hardware implementations: A Survey," by Xin Fang, 2019

Today there are three main classes of hardware available for model deployment with various trade-offs between flexibility and efficiency:

- **Standard programmable CPU/GPUs with specialised co-processors:** A common approach is the use of specialised co-processors, which can be either directly co-located with the general-purpose processor on the same silicon or might be connected as a separate chip. These accelerators stretch the power continuum from relatively simple digital signal processors (DSPs) to highly parallel matrix computation units and similar advanced designs. These accelerators can either be monolithic designs such as special Edge variants of Google's TPU or a distributed set of smaller compute cores. Some examples for the latter are the Tensor Cores in newer Nvidia GPU architectures, the Hexagon cores in Qualcomm SnapDragon SoCs and Intel Movidius SHAVE processors. They commonly need specialised drivers and software libraries that allow software developers to take advantage of their capabilities.

- **Digital hardware:** For greater efficiency, field-programmable gate arrays (FPGAs) can be combined with powerful CPU systems, specialised arrays of AI accelerator cores and traditional fabric, where the hardware can be programmed by the use of hardware description language (HDL) or C/C++ via high-level synthesis tools. FPGAs combine the benefits of specialised hardware with the freedom to change the layout even after the chip has left the factory. The use of FPGA can create more dynamic, scalable, and flexible systems, even though they often carry higher cost. Another type of processor is emerging as a new class of processing accelerator for these predominantly data-centric heterogeneous processing tasks as offload to the main CPU. These processors are a well-partitioned circuit that comprises all the control and arithmetic logic components necessary to execute machine learning

algorithms and are referred as NPUs (Neural Processing Unit). NPUs are designed to accelerate the performance of common machine learning tasks. NPUs may be part of a large SoC, a plurality of NPUs may be instantiated on a single chip, or they may be part of a dedicated neural-network accelerator.

- **Analog hardware**: The biggest time and energy costs in most computers occur when lots of data must move between external memory and computational resources such as CPUs, GPUs, or NPUs. This is the "von Neumann bottleneck," named after the classic computer architecture that separates memory and logic. One way to greatly reduce the power needed for machine learning is to avoid moving the data — to do the computation where the data is located. Because there is no movement of data, tasks can be performed in a fraction of the time and require much less energy. In-memory and neuromorphic analogue computing are such approaches that take this approach. In-memory computing is the design of memories next to or within the processing elements of hardware such that bitwise operations and arithmetic operations can occur in memory. Neuromorphic analogue computing allows in-place compute but also mimics the brain's function and efficiency by building artificial neural systems that implement «neurons» and «synapses» to transfer electrical signals via an analogue circuit design. This circuit is the breakthrough technology solution to the Von Neumann bottleneck problem. Analog neuromorphic ICs are intrinsically parallel, and better adapted for neural network operations than current digital hardware solutions offering orders of magnitude improvements for edge applications. Market-ready, end-user programmable chips are an essential need for neuromorphic computing to expand its visibility and to achieve a variety of "real-world applications" with an increasing number of users. As large technology companies are waiting for the technology to become more mature, some start-ups have released their chips to fill the gap and to have a competitive advantage against those tech-giants. Examples include BrainChip AKIDA, GrAI Matter Labs VIP, and SynSense DYNAP processors.

Software for Edge AI

Edge computing software requires the implementation of comprehensive intelligent edge frameworks and platforms addressing specific requirements and challenges spanning hardware, power efficiency, software, connectivity, flexibility and interoperability, and security. Software stacks for edge computing are inherently fragmented due to the following:

- Not all applications are built the same way, the software stack will determine how well the final application will perform.
- To achieve full capacity for an application, a software stack needs to be tailored specifically to what is trying to be accomplished and the underlying hardware.
- Different models require specific tools that only customizable stacks will offer.
- The software stack must include a scheduler and run-time engine that must meet varying requirements of latency-sensitive applications.
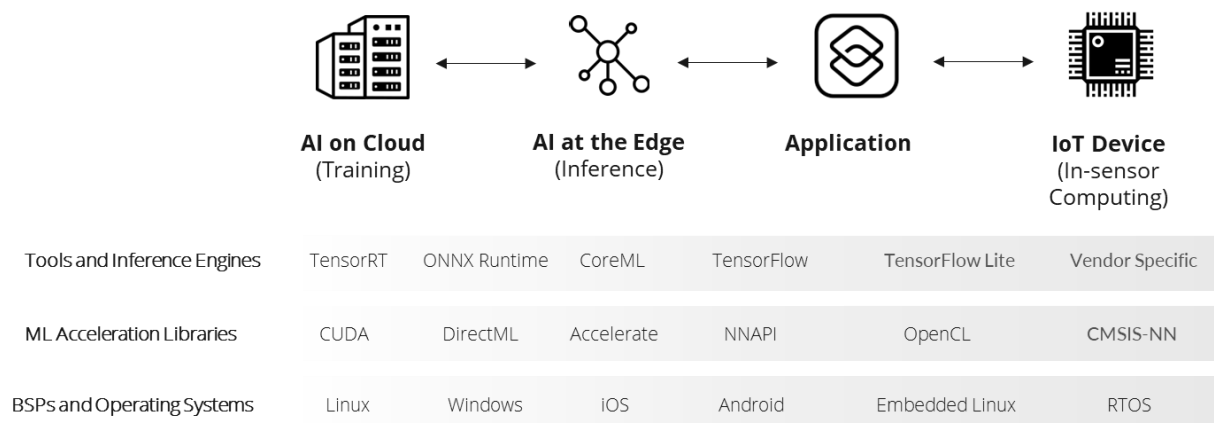
Software for machine learning can be broken down into three main categories:

		Form Factors	Algorithms	Hardware	
	Cloud ML	Data Centers Super Computers	Deep neural network	TPU, FPGA, GPU, and CPU	**CPU:** x86/ARM64 **Memory:** TB **Storage:** PB **Clock Freq:** GHz **Power:** 250W
	Edge ML	Local Servers Personal Computers Smart Phones Smart Devices	Optimized algorithms and lightweight convolution neural networks	ASICs with NPU / Coprocessors	**CPU:** ARM **Memory:** MB - GB **Storage:** GB - TB **Clock Freq:** MHz - GHz **Power:** ~8W
	Tiny ML	Wearables Smart Sensors Microcontrollers Actuators	Micro convolution neural networks	MCUs with Neuromorphic Fabric	**CPU:** MCU **Memory:** ~500KB **Storage:** <2MB **Clock Freq:** <200MHz **Power:** <1W

- **CloudML:** In CloudML, machine learning is used to integrate artificial intelligence into applications via cloud infrastructure. Cloud service providers provide a wide range of AI services that developers can make use of in their applications. Devices that transmit the data required by these services are connected to the internet through which they send the data for processing, storage and analysis. CloudML can be characterised by:
  - Typically applied to a limited number of focused, vertical applications
  - Targeting a small range of processors
  - Plenty of available power and bandwidth
  - Large equipment budget

- **EdgeML**: EdgeML is made for real-time, always-on solutions. By processing data as close to its source as possible, latency is minimized and organizations gain actionable insights in real time. EdgeML can be characterised by:
  - Potentially applied to a wide and diverse range of applications
  - Many possible processor targets, from CPUs and GPUs to NPUs, DSPs and other forms of dedicated accelerator
  - Numerous – often proprietary – application programming interfaces (APIs)
  - Relatively low-cost devices operating in thermally and power-constrained environments

- **TinyML:** TinyML is a fast-growing multidisciplinary field at the intersection of machine learning, hardware, and software, that focuses on enabling deep learning algorithms on embedded (microcontroller powered) devices operating at extremely low power range (mW range and below) targeting in-sensor computing. TinyML can be characterised by:
  - compact and low-cost devices (microcontrollers)
  - very low power usage
  - extremely limited memory capacity
  - low lag time (almost immediate) integrated machine learning algorithms analysis.

Moving AI from the cloud to the edge can be challenging because of the need to create neural network architectures that can operate well in edge AI chips. Cloud servers run on general computing platforms which can perform on any network architecture, but in edge AI, the models and architectures must be modified to cope with the low resources of the AI modules at the edge. In

addition, tools, inference engines, acceleration frameworks, and operating systems vary greatly from the cloud, edge computing, and IoT devices.

	AI on Cloud (Training)		**AI at the Edge** (Inference)	**Application**		**IoT Device** (In-sensor Computing)
Tools and Inference Engines	TensorRT	ONNX Runtime	CoreML	TensorFlow	TensorFlow Lite	Vendor Specific
ML Acceleration Libraries	CUDA	DirectML	Accelerate	NNAPI	OpenCL	CMSIS-NN
BSPs and Operating Systems	Linux	Windows	iOS	Android	Embedded Linux	RTOS

To efficiently leverage this heterogeneous computing environment for edge computing, it is critical to have a full software stack able to manage multiple layers from high-level level training, through efficient inference engines, to developer board support packages and operating systems. The main components being:

- **Datatools** are containerized python-based utilities, which extract data from a given source dataset and translate both unstructured (images, audio, video files etc.) and semi-structured (CSV, JSON, TXT, XML, XLS, etc.) data to common representation. Datatools generate a standard data structure and definitions for the annotations, which make data easier to analyse, consume and reuse by the subsequent model training pipeline.
- **Training Pipelines** are workflows that deliver powerful and easy-to-deploy building blocks for creating complex AI models that can be deployed on cyber-physical systems. By taking care of many end-to-end tooling dependencies and providing standardised interfaces, the Training Pipelines enable users to focus on producing optimal solutions while allowing faster feedback during the implementation of end user requirements. Such workflows enable accelerated deployment of deep learning models to resource constrained low-power embedded systems (Deep Edge).
- **Runtime Frameworks** provide the capabilities to generate portable and efficient implementations of DNNs that can be deployed and optimised across heterogeneous platforms, e.g., CPU, GPU, FPGA, DSP, NPU. Such frameworks supply a collection of tools, executables, libraries and inference engines, featuring a full development flow for deep learning solutions on embedded devices by providing platform support, sample models, optimisation tools, and benchmarking.
- **Hardware Acceleration Libraries** are the collection of libraries and backends that allows the actual execution of the neural network on the hardware device. Hardware Acceleration Libraries are tightly coupled with the underlying hardware processor or accelerator, e.g., CPU, GPU, NPU, to fully optimise the AI workload on the device. General Hardware Acceleration Libraries may vary severely from system to system, often requiring a design search exploration to find the optimal one for a given target platform.
- **Board Support and Manager Packages** contain the full software stack (operating system, drivers, middleware components, etc.) and documentation required to procure, set up and

control target hardware for the execution of AI Applications in an easy and reproducible format. The use of a Manager Package which works as a running environment for AI algorithms on the edge platform, supports inference tasks, benchmarking, and model re-training, easing the usage of AI on such difficult and heterogeneous environments.

References

Abdo, A. A., Lawgali, A. O., & Abdalla, M. (2020). A Review of the Iris Recognition Methods Used for the Individual Authentication. *Transactions on Machine Learning and Artificial Intelligence*, *8*(6), 16–27. https://doi.org/10.14738/tmlai.86.9687

Albadawi, Y., Takruri, M., & Awad, M. (2022). A Review of Recent Developments in Driver Drowsiness Detection Systems. *Sensors*, *22*(5), 2069. https://doi.org/10.3390/s22052069

Andriluka, M., Pishchulin, L., Gehler, P., & Schiele, B. (2014, June). 2D Human Pose Estimation: New Benchmark and State of the Art Analysis. *2014 IEEE Conference on Computer Vision and Pattern Recognition*. http://dx.doi.org/10.1109/cvpr.2014.471

Ashmawi, S., Alharbi, M., Almaghrabi, A., & Alhothali, A. (2019). FITME: BODY MEASUREMENT ESTIMATIONS USING MACHINE LEARNING METHOD. *Procedia Computer Science*, *163*, 209–217. https://doi.org/10.1016/j.procs.2019.12.102

Bandini, A., & Zariffa, J. (2020). Analysis of the hands in egocentric vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–1. https://doi.org/10.1109/tpami.2020.2986648

Berthelier, A., Chateau, T., Duffner, S., Garcia, C., & Blanc, C. (2020). Deep Model Compression and Architecture Optimization for Embedded Systems: A Survey. *Journal of Signal Processing Systems*, *93*(8), 863–878.

Brown, S. (n.d.). *Machine learning, explained*. MIT Sloan. Retrieved June 24, 2022, from https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained

Bruha, I. (2001). Pre- and Post-processing in Machine Learning and Data Mining. In *Machine Learning and Its Applications* (pp. 258–266). Springer Berlin Heidelberg. http://dx.doi.org/10.1007/3-540-44673-7_13

Cañigueral, R., & Hamilton, A. F. de C. (2019). The Role of Eye Gaze During Natural Social Interactions in Typical and Autistic People. *Frontiers in Psychology*, *10*. https://doi.org/10.3389/fpsyg.2019.00560

Chan, Y., Talburt, J., & Talley, T. M. (2009). *Data Engineering: Mining, Information and Intelligence*. Springer Science & Business Media

Chang, J. Y., Moon, G., & Lee, K. M. (2018, June). V2V-PoseNet: Voxel-to-Voxel Prediction Network for Accurate 3D Hand and Human Pose Estimation from a Single Depth Map. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Chevtchenko, S. F., Vale, R. F., Macario, V., & Cordeiro, F. R. (2018). A convolutional neural network with feature fusion for real-time hand posture recognition. *Applied Soft Computing*, *73*, 748–766. https://doi.org/10.1016/j.asoc.2018.09.010

Chhatkuli, S., Jiang, I., & Kamiyama, K. (2022, May 23). *Body Composition Estimation Based on Multimodal Multi-task Deep Neural Network*. arXiv.Org. https://arxiv.org/abs/2205.11031

Christensen, D. V., Dittmann, R., Linares-Barranco, B., Sebastian, A., Gallo, M. L., Redaelli, A., Slesazeck, S., Mikolajick, T., Spiga, S., Menzel, S., Valov, I., Milano, G., Ricciardi, C., Liang, S.-J., Miao, F., Lanza, M., Quill, T. J., Keene, S. T., Salleo, A., … Pryds, N. (2021, May 12). *2022 Roadmap on Neuromorphic Computing and Engineering*. arXiv.Org. https://arxiv.org/abs/2105.05956

Collins, G. R. (2020). Improving human–robot interactions in hospitality settings. *International Hospitality Review*, *34*(1), 61–79. https://doi.org/10.1108/ihr-09-2019-0019

Coral TPU. (n.d.). Coral. Retrieved June 29, 2022, from https://coral.ai/products/

Correll, N. (n.d.). *Introduction to Autonomous Robots*. Retrieved June 24, 2022, from ttps://open.umn.edu/opentextbooks/textbooks/316

de Prado, M. (n.d.). *Bonseyes AI Pipeline—Bringing AI to You: End-to-end integration of data, algorithms, and deployment tools*. ACM Digital Library. https://dl.acm.org/doi/abs/10.1145/3403572

Dooley, S., Downing, R., Wei, G., Shankar, N., Thymes, B., Thorkelsdottir, G., Kurtz-Miott, T., Mattson, R., Obiwumi, O., Cherepanova, V., Goldblum, M., Dickerson, J. P., & Goldstein, T. (2021, October 15). *Comparing Human and Machine Bias in Face Recognition*. arXiv.Org. https://arxiv.org/abs/2110.08396

Dubey, P. (2020, April 28). *Virtuous Cycle of AI: System-level Challenges*. https://r6.ieee.org/scv-eps/?p=1139

Durmaz, T. B. (n.d.). *Neuromorphic*. *Edge Computing Market Size, Share & Growth Report, 2030*. (n.d.). Retrieved June 29, 2022, from https://www.grandviewresearch.com/industry-analysis/edge-computing-market

Ekman, P. (1999). Emotional and conversational nonverbal signals. In *Gesture, Speech, and Sign* (pp. 45–55). Oxford University Press.

Ekman, P., & Rosenberg, E. L. (2005). *What the Face Reveals: Basic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System (FACS)*. Oxford University Press.

Escudero, M. de P. (2021, July 19). *On automation for optimised and robust deployment of neural networks on edge devices*. Research Collection. https://www.research-collection.ethz.ch/handle/20.500.11850/495609

Euro NCAP. (n.d.). *Euro NCAP Road Map 2025 - In Pursuit of Vision Zero*. Euro NCAP Road Map 2025 - In Pursuit of Vision Zero. Retrieved June 24, 2022, from https://www.euroncap.com/en/press-media/press-releases/euro-ncap-launches-road-map-2025-in-pursuit-of-vision-zero/

Farah, M. J., Wilson, K. D., Drain, M., & Tanaka, J. N. (1998). What is "special" about face perception? *Psychological Review*, *105*(3), 482–498. https://doi.org/10.1037/0033-295x.105.3.482

Feng, P., Yu, M., Naqvi, S. M., & Chambers, J. A. (2014, August). Deep learning for posture analysis in fall detection. *2014 19th International Conference on Digital Signal Processing*. http://dx.doi.org/10.1109/icdsp.2014.6900806

Governance, H. S., & UHL, F. (2021, December 15). Global expenditure on health: Public spending on the rise? *World Health Organization*. https://www.who.int/publications/i/item/9789240041219

Gu, F., Chung, M.-H., Chignell, M., Valaee, S., Zhou, B., & Liu, X. (2022). A Survey on Deep Learning for Human Activity Recognition. *ACM Computing Surveys*, *54*(8), 1–34. https://doi.org/10.1145/3472290

Ionescu, C., Papava, D., Olaru, V., & Sminchisescu, C. (2014). Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *36*(7), 1325–1339. https://doi.org/10.1109/tpami.2013.248

Ivanov, D., Chezhegov, A., Grunin, A., Kiselev, M., & Larionov, D. (2022, May 25). *Neuromorphic artificial intelligence systems*. arXiv.Org. https://arxiv.org/abs/2205.13037

Jongerius, C., Hessels, R. S., Romijn, J. A., Smets, E. M. A., & Hillen, M. A. (2020). The Measurement of Eye Contact in Human Interactions: A Scoping Review. *Journal of Nonverbal Behavior*, *44*(3), 363–389. https://doi.org/10.1007/s10919-020-00333-3

Kaisler, R. E., Marin, M. M., & Leder, H. (2020). Effects of Emotional Expressions, Gaze, and Head Orientation on Person Perception in Social Situations. *SAGE Open*, *10*(3), 215824402094070. https://doi.org/10.1177/2158244020940705

Kalita, D. (2022, March 17). *A Quick Overview of Data Engineering*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2022/03/a-quick-overview-of-data-engineering/

Kshirsagar, P. R., Dadheech, P., Yuvaraj, T., Moorthy, C. A. S., & Upadhyaya, M. (2022). Fatigue detection using artificial intelligence. *RECENT TRENDS IN SCIENCE AND ENGINEERING*. http://dx.doi.org/10.1063/5.0074121

Kulikajevas, A., Maskeliunas, R., & Damaševičius, R. (2021). Detection of sitting posture using hierarchical image composition and deep learning. *PeerJ Computer Science*, *7*, e442. https://doi.org/10.7717/peerj-cs.442

Larcher, J., & DeJohn, C. A. (n.d.). *Incapacitation Data Registry History and Development*.

Lee, Cho, Lee, & Whang. (2019). Vision-Based Measurement of Heart Rate from Ballistocardiographic Head Movements Using Unsupervised Clustering. *Sensors*, *19*(15), 3263. https://doi.org/10.3390/s19153263

Lee, S. U. (2021). *Cognitive Human Activity and Plan Recognition for Human-Robot Collaboration* [Thesis, MIT].

Li, J., Zhang, L., Guo, D., Zhuo, S., & Sim, T. (2015, May). Audio-visual twins database. *2015 International Conference on Biometrics (ICB)*. http://dx.doi.org/10.1109/icb.2015.7139115

Liang, J., Zou, Y.-Q., Liang, S.-Y., Wu, Y.-W., & Yan, W.-J. (2021). Emotional Gaze: The Effects of Gaze Direction on the Perception of Facial Emotions. *Frontiers in Psychology*, *12*. https://doi.org/10.3389/fpsyg.2021.684357

Liu, D., Kong, H., Luo, X., Liu, W., & Subramaniam, R. (2022). Bringing AI to edge: From deep learning's perspective. *Neurocomputing*, *485*, 297–320. https://doi.org/10.1016/j.neucom.2021.04.141

Lopatina, O. L., Komleva, Y. K., Gorina, Y. V., Higashida, H., & Salmina, A. B. (2018). Neurobiological Aspects of Face Recognition: The Role of Oxytocin. *Frontiers in Behavioral Neuroscience*, *12*. https://doi.org/10.3389/fnbeh.2018.00195

ml-ops.org. (n.d.). Retrieved June 29, 2022, from https://ml-ops.org/content/end-to-end-ml-workflow#model-engineering

Mota, T., Sridharan, M., & Leonardis, A. (2021). Integrated Commonsense Reasoning and Deep Learning for Transparent Decision Making in Robotics. *SN Computer Science*, *2*(4). https://doi.org/10.1007/s42979-021-00573-0

Mukhopadhyay, S. C., Tyagi, S. K. S., Suryadevara, N. K., Piuri, V., Scotti, F., & Zeadally, S. (2021). Artificial Intelligence-Based Sensors for Next Generation IoT Applications: A Review. *IEEE Sensors Journal*, *21*(22), 24920–24932.

Mutlu, O. (2018, June). Processing data where it makes sense in modern computing systems: Enabling in-memory computation. *2018 7th Mediterranean Conference on Embedded Computing (MECO)*. http://dx.doi.org/10.1109/meco.2018.8405955

NeurIPS Data-Centric AI Workshop. (n.d.). Retrieved June 24, 2022, from https://datacentricai.org/neurips21/

Norton, E. (n.d.). *Science*. AAAS. Retrieved June 16, 2022, from https://www.science.org/content/article/identifying-brains-own-facial-recognition-system

Odekerken-Schröder, G., Mele, C., Russo-Spena, T., Mahr, D., & Ruggiero, A. (2020). Mitigating loneliness with companion robots in the COVID-19 pandemic and beyond: an integrative framework and research agenda. *Journal of Service Management*, *31*(6), 1149–1162. https://doi.org/10.1108/josm-05-2020-0148

O'Toole, A. J., Jonathon Phillips, P., Jiang, F., Ayyad, J., Penard, N., & Abdi, H. (2007). Face Recognition Algorithms Surpass Humans Matching Faces Over Changes in Illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *29*(9), 1642–1646. https://doi.org/10.1109/tpami.2007.1107

Otsuka, K., & Tsumori, M. (2020). Analyzing Multifunctionality of Head Movements in Face-to-Face Conversations Using Deep Convolutional Neural Networks. *IEEE Access*, *8*, 217169–217195. https://doi.org/10.1109/access.2020.3041672

Petrovic, V. M. (2018). Artificial Intelligence and Virtual Worlds – Toward Human-Level AI Agents. *IEEE Access*, *6*, 39976–39988. https://doi.org/10.1109/access.2018.2855970

Piñero-Fuentes, E., Canas-Moreno, S., Rios-Navarro, A., Domínguez-Morales, M., Sevillano, J. L., & Linares-Barranco, A. (2021). A Deep-Learning Based Posture Detection System for Preventing Telework-Related Musculoskeletal Disorders. *Sensors*, *21*(15), 5236. https://doi.org/10.3390/s21155236

Products. (n.d.). Coral. Retrieved June 29, 2022, from https://coral.ai/products/

Publication of the ECS SRIA 2022. (2022, January 17). Aeneas. https://aeneas-office.org/2022/01/17/publication-of-ecs-sria-2022/

Ramseyer, F., & Tschacher, W. (2014). Nonverbal synchrony of head- and body-movement in psychotherapy: different signals have different associations with outcome. *Frontiers in Psychology*, *5*. https://doi.org/10.3389/fpsyg.2014.00979

Roh, Y., Heo, G., & Whang, S. E. (2021). A Survey on Data Collection for Machine Learning: A Big Data - AI Integration Perspective. *IEEE Transactions on Knowledge and Data Engineering*, *33*(4), 1328–1347. https://doi.org/10.1109/tkde.2019.2946162

Romano, C., Schena, E., Silvestri, S., & Massaroni, C. (2021). Non-Contact Respiratory Monitoring Using an RGB Camera for Real-World Applications. *Sensors*, *21*(15), 5126. https://doi.org/10.3390/s21155126

Roter, D. L., Frankel, R. M., Hall, J. A., & Sluyter, D. (2006). The Expression of Emotion Through Nonverbal Behavior in Medical Visits. Mechanisms and Outcomes. *Journal of General Internal Medicine*, *21*(S1), S28–S34.

Safford, A., Kegel, J., Hershaw, J., Girard, D., & Ettenhofer, M. (2015). Eye-Tracking Technology for Estimation of Cognitive Load After Traumatic Brain Injury. In *Foundations of Augmented Cognition* (pp. 136–143). Springer International Publishing.

Sarker, I. H. (2022). AI-Based Modeling: Techniques, Applications and Research Issues Towards Automation, Intelligent and Smart Systems. *SN Computer Science*, *3*(2). https://doi.org/10.1007/s42979-022-01043-x

Shafique, M., Theocharides, T., Bouganis, C.-S., Hanif, M. A., Khalid, F., Hafiz, R., & Rehman, S. (2018, March). An overview of next-generation architectures for machine learning: Roadmap, opportunities and challenges in the IoT era. *2018 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. http://dx.doi.org/10.23919/date.2018.8342120

Sharma, A., Sharma, V., Jaiswal, M., Wang, H.-C., Jayakody, D. N. K., Basnayaka, C. M. W., & Muthanna, A. (2022). Recent Trends in AI-Based Intelligent Sensing. *Electronics*, *11*(10), 1661. https://doi.org/10.3390/electronics11101661

Sharma, R. P., & Verma, G. K. (2015). Human Computer Interaction using Hand Gesture. *Procedia Computer Science*, *54*, 721–727.

Shimonishi, K., & Kawashima, H. (2020). A two-step approach for interest estimation from gaze behavior in digital catalog browsing. *Journal of Eye Movement Research*, *13*(1). https://doi.org/10.16910/jemr.13.1.4

Smith, E., Storch, E. A., Vahia, I., Wong, S. T. C., Lavretsky, H., Cummings, J. L., & Eyre, H. A. (2021). Affective Computing for Late-Life Mood and Cognitive Disorders. *Frontiers in Psychiatry*, *12*. https://doi.org/10.3389/fpsyt.2021.782183

Sorci, M., Antonini, G., Cruz, J., Robin, T., Bierlaire, M., & Thiran, J.-Ph. (2010). Modelling human perception of static facial expressions. *Image and Vision Computing*, *28*(5), 790–806. https://doi.org/10.1016/j.imavis.2009.10.003

Stedman, C., & McLaughlin, E. (2022, February 2). Data Collection. *TechTarget*. https://www.techtarget.com/searchcio/definition/data-collection

Šumak, B., Brdnik, S., & Pušnik, M. (2021). Sensors and Artificial Intelligence Methods and Algorithms for Human–Computer Intelligent Interaction: A Systematic Mapping Study. *Sensors*, *22*(1). https://doi.org/10.3390/s22010020

Sumiya, T., Matsubara, Y., Nakano, M., & Sugaya, M. (2015). A Mobile Robot for Fall Detection for Elderly-Care. *Procedia Computer Science*, *60*, 870–880. https://doi.org/10.1016/j.procs.2015.08.250

Synced. (2017, April 28). Artificial Intelligence is the New Electricity — Andrew Ng. *SyncedReview*. https://medium.com/syncedreview/artificial-intelligence-is-the-new-electricity-andrew-ng-cc132ea6264

Szymkowski, M., Jasiński, P., & Saeed, K. (2021). Iris-based human identity recognition with machine learning methods and discrete

fast Fourier transform. *Innovations in Systems and Software Engineering*, *17*(3), 309–317.

Taborri, J., Molinaro, L., Santospagnuolo, A., Vetrano, M., Vulpiani, M. C., & Rossi, S. (2021). A Machine-Learning Approach to Measure the Anterior Cruciate Ligament Injury Risk in Female Basketball Players. *Sensors*, *21*(9), 3141.

tinyML Foundation. (n.d.). tinyML Foundation. Retrieved June 29, 2022, from https://www.tinyml.org/

Toshpulatov, M., Lee, W., Lee, S., & Haghighian Roudsari, A. (2022). Human pose, hand and mesh estimation using deep learning: a survey. *The Journal of Supercomputing*, *78*(6), 7616–7654. https://doi.org/10.1007/s11227-021-04184-7

Tripathi, Jalal, & Agrawal. (2017). Suspicious human activity recognition: a review. *Artificial Intelligence Review*, *50*(2), 283–339.

University of Minnesota. (2015, October 26). *4.2 Seeing*. University of Minnesota Libraries Publishing Edition, 2015. This Edition Adapted from a Work Originally Produced in 2010 by a Publisher Who Has Requested That It Not Receive Attribution. . https://open.lib.umn.edu/intropsyc/chapter/4-2-seeing/

Varol, G., Ceylan, D., Russell, B., Yang, J., Yumer, E., Laptev, I., & Schmid, C. (2018, April 13). *BodyNet: Volumetric Inference of 3D Human Body Shapes*. arXiv.Org. https://arxiv.org/abs/1804.04875

Vehlen, A., Spenthof, I., Tönsing, D., Heinrichs, M., & Domes, G. (2021). Evaluation of an eye tracking setup for studying visual attention in face-to-face conversations. *Scientific Reports*, *11*(1). https://doi.org/10.1038/s41598-021-81987-x

Vignolo, A., Noceti, N., Rea, F., Sciutti, A., Odone, F., & Sandini, G. (2017). Detecting Biological Motion for Human–Robot Interaction: A Link between Perception and Action. *Frontiers in Robotics and AI*, *4*. https://doi.org/10.3389/frobt.2017.00014

Walambe, R., Nayak, P., Bhardwaj, A., & Kotecha, K. (2021). Employing Multimodal Machine Learning for Stress Detection. *Journal of Healthcare Engineering*, *2021*, 1–12. https://doi.org/10.1155/2021/9356452

Wang, M., & Deng, W. (2021). Deep face recognition: A survey. *Neurocomputing*, *429*, 215–244.

Wang, Y., Wang, J., Zhang, W., Zhan, Y., Guo, S., Zheng, Q., & Wang, X. (2022). A survey on deploying mobile deep learning applications: A systemic and technical perspective. *Digital Communications and Networks*, *8*(1), 1–17.

Weiner, I. B., Healy, A. F., & Proctor, R. W. (2012). *Handbook of Psychology, Experimental Psychology* (pp. 85–119). John Wiley & Sons.

Weiss, A., Wortmeier, A.-K., & Kubicek, B. (2021). Cobots in Industry 4.0: A Roadmap for Future Practice Studies on Human–Robot Collaboration. *IEEE Transactions on Human-Machine Systems*, *51*(4), 335–345. https://doi.org/10.1109/thms.2021.3092684

What is hardware-in-the-loop testing? (n.d.). Aptiv. Retrieved June 29, 2022, from https://www.aptiv.com/en/insights/article/what-is-hardware-in-the-loop-testing

Why the Arm Ethos-U65 microNPU is a big deal, and how it came to be this way. (n.d.). NXP Semiconductors. Retrieved June 29, 2022, from https://www.nxp.com/company/blog/why-the-arm-ethos-u65-micronpu-is-a-big-deal-and-how-it-came-to-be-this-way:BL-ARM-ETHOS-U65-MICRONPU

World Health Organization. (n.d.). *Gesture Recognition Market Size & Share Report, 2030*. Retrieved June 20, 2022, from https://www.grandviewresearch.com/industry-analysis/gesture-recognition-market

Woźniak, S., Pantazi, A., Bohnstingl, T., & Eleftheriou, E. (2020). Deep learning incorporating biologically inspired neural dynamics and in-memory computing. *Nature Machine Intelligence*, *2*(6), 325–336. https://doi.org/10.1038/s42256-020-0187-0

Wrobel, L. (2022, April 29). *"Data-centric AI" – A shift in the AI mindset?* Ccecosystems.News. https://ccecosystems.news/en/data-centric-ai-a-shift-in-the-ai-mindset/

Xing, Y., Lv, C., Wang, H., Cao, D., Velenis, E., & Wang, F.-Y. (2019). Driver Activity Recognition for Intelligent Vehicles: A Deep Learning Approach. *IEEE Transactions on Vehicular Technology*, *68*(6), 5379–5390. https://doi.org/10.1109/tvt.2019.2908425

Yasen, M., & Jusoh, S. (2019). A systematic review on hand gesture recognition techniques, challenges and applications. *PeerJ Computer Science*, *5*, e218. https://doi.org/10.7717/peerj-cs.218

Yu, C.-P. (2021, December 13). *How to develop efficient AI for edge computing*. AI Business. https://aibusiness.com/author.asp?section_id=796&doc_id=774036

Zagermann, J., Pfeil, U., & Reiterer, H. (2016). Measuring Cognitive Load using Eye Tracking Technology in Visual Computing. *Proceedings of the Beyond Time and Errors on Novel Evaluation Methods for Visualization - BELIV '16*.

ZF: AI-in-the-Loop. (n.d.). dSPACE. Retrieved June 29, 2022, from https://www.dspace.com/en/inc/home/applicationfields/stories/zf-ai-in-the-loop.cfm

Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C.-L., & Grundmann, M. (2020, June 18). *MediaPipe Hands: On-device Real-time Hand Tracking*. arXiv.Org. https://arxiv.org/abs/2006.10214

Zhang, Y., Yang, X., & Feng, Z. (2021). Driving distraction detection based on gaze activity. *Electronics Letters*, *57*(22), 857–859.

Zhang, Z., Wen, F., Sun, Z., Guo, X., He, T., & Lee, C. (2022). Artificial Intelligence-Enabled Sensing Technologies in the 5G/Internet of Things Era: From Virtual Reality/Augmented Reality to the Digital Twin. *Advanced Intelligent Systems*, 2100228.

Zheng, C., Wu, W., Chen, C., Yang, T., Zhu, S., Shen, J., Kehtarnavaz, N., & Shah, M. (2020, December 24). *Deep Learning-Based Human Pose Estimation: A Survey*. arXiv.Org. https://arxiv.org/abs/2012.13392

Zhou, Z., Chen, X., Li, E., Zeng, L., Luo, K., & Zhang, J. (2019). Edge Intelligence: Paving the Last Mile of Artificial Intelligence With Edge Computing. *Proceedings of the IEEE*, *107*(8), 1738–1762. https://doi.org/10.1109/jproc.2019.2918951

Zuliani, P. (2020, July 27). *The Long-term Advantages of an AI-Powered Virtuous Circle*. Pegus Digital. https://pegus.digital/the-long-term-advantages-of-an-ai-powered-virtuous-circle/

About NVISO

NVISO is an Artificial Intelligence company founded in 2009 and headquartered at the Innovation Park of the École Polytechnique Fédérale de Lausanne (EPFL) in Switzerland with offices in Serbia and Japan. Its mission is to help teach machines to understand people and their behaviour to make autonomous machines safe, secure, and personalized for humans. As leader in human behavioural AI, it provides robust embedded software solutions that can sense, comprehend, and act upon human behaviour in real-world environments deployed at the deep edge. It achieves this through real-time perception and observation of people and objects in contextual situations combined with the reasoning and semantics of human behaviour based on trusted scientific research. NVISO's technology is made accessible through ready-to-use AI solutions addressing Smart Mobility and Smart Health and Living applications (in-cabin monitoring systems, health assessments, and companion robot sensing) with a key focus on the deep and extreme edge with ultra-low power processing capabilities. With a singular focus on how to apply the most advanced and robust technology to industry and societal problems that matter, NVISO's solutions help advance human potential through more robust and rich human machine interactions. www.nviso.ai