Volume 1.0

Lacueva Pérez F. J.
Brandl, P.
Mayo Macías S.
Gracia Bandrés, M.A.
Romero Martín, D.

Project Deliverable 2.1

Technology Monitoring: Report on Information Needed For Workers in the Smart Factory

Worker-Centric Workplaces in Smart Factories

www.facts4workers.eu

FACTS4WORKERS

Factories of the Future
Public Private Partnership

Series: Heading
Published by: FACTS4WORKERS: Worker-Centric Workplaces in Smart Factories.
FoF 2014/636778

Volume 1.0: Technology Monitoring: Report on Information Needed For the Industrial Challenges Workers with Taxonomy

Reference / Citation

Lacueva Perez, F. J., Brandl, P., Mayo Macias, S. Gracia Bandrés, M.A., Romero Martín, D. (2015) Project Report – FACTS4WORKERS: Worker-Centric Workplace in Smart Factories

www.facts4workers.eu

1. Printing, November 2015
Cover Design: Florian Ott, Cooperation Systems Center Munich

Worker-Centric Workplaces in Smart Factories

E-Mail: facts4workers@v2c2.at
Internet: www.facts4workers.eu

About this document

Executive Summary

D2.1, Technology Monitoring: Report on Information Needed for the Industrial Challenges Workers with Taxonomy is part of the work in progress of the "FACTorieS for WORKERS" (FACTS4WORKERS) project and specifically of the T2.1 task of WP2.

Developing the worker-centric HCI/HMI building blocks

WP2, Worker-centric HCI/HMI Building Blocks, aims to develop (in a co-creation process with shop-floor staff) the smart factory solution's worker-centric HCI/HMI building blocks, characterised by maximum usability, user experience (UX) and technology acceptance. As a preliminary work, or expressed as "parallel guiding work", Task 2.1, Analysis of Technical Requirements and Technology Monitoring will create and maintain a state of the art regarding available and trending technologies (devices, software developments and tools etc.) within the very dynamic field of today's HCI technology (smart glasses, smart textiles etc.). Furthermore, Task 2.1 will review new (disrupting) HCI paradigms and will relate them to already established (and sometimes outdated) HCI paradigms.

Creating a vision of current and future developments of HCI technologies and paradigms

D2.1 is the result of the work of T2.1. Its final objective is to create a vision of the current and future developments of HCI technologies and paradigms that will allow other WP2 tasks to obtain the maximum benefit when implementing HCI building blocks, as well as support future technologies adaptation as they become available during the project execution. D2.1 will also provide a general evaluation of existing technologies considering their applicability on the factories' shop floor but always observing project objectives and industrial challenges reflected in the project proposal. The technologies evaluation will be provided as a taxonomy of technologies that will be evaluated on a TRL-based scale. The taxonomy will be updated in subsequent versions in order to track the technology maturity evolution during the project life. It will also comment on the observed state of technology.

Keywords

HCI critical taxonomy, Industry 4.0, Factory of The Future, Shop floor, Smart devices, Wearables, Augmented Reality.

Document authors and reviewers

The following individuals have made a direct contribution to the document. Please note that many others have also supported our work, and we thank them all sincerely.

Lead Authors

Name	Organisation	Role
Francisco J. Lacueva	ITA	WP2
Peter Brandl	EVO	WP2 – Leader

Featuring Authors

Name	Organisation	Role
M. A. Gracia	ITA	WP2
S. Mayo	ITA	WP2
D. Romero	ITA	WP2
M. A. Gracia	ITA	WP2

Reviewers

Name	Organisation	Role
Martin Wifling	ViF	Project Coordinator
Alexander Richter	UZH	WP1 – Leader
Gianni Campatelli	UFI	WP3 – Leader

Table of Contents

Table of Figures

Index of Abbreviations

App .... Application

CP .... Cross-Platform

CSCW.................. Computer-Supported Cooperative Work

CV .... Computer Vision

DOF .... Degrees of Freedom

DSL...................... Domain-Specific Language

GR........................ Gesture Recognition

GUI .... Graphic User Interface

HA .... Hybrid Apps

HCI .... Human–Computer Interaction

HWR.................... Handwriting Recognition

IR .... Infrared

IMU .... Inertial Measurement Unit

IoT........................ Internet of Things

K-12..................... Term used to refer to the sum of primary and secondary education

LBS....................... Location-Based Service

LTM .... Long-Term Memory

MWP.................... Mobility Web Page

NA........................ Native Apps

NUI .... Natural User Interface

OS .... Operating System

OSS...................... Open Source Software

OST...................... Optical See-Through

QoE...................... Quality of Experience

QR........................ Quick Response (codes)

QS .... Quantified Self

RF .... Radio Frequency

RFID.................... Radio Frequency Identification

RSSI..................... Received Signal Strength Information

RVC .... Reality–Virtuality Continuum

SAW .... Surface Acoustic Wave

SDK .... Software Development Kits

SLAM .... Simultaneous Localisation and Mapping

SNS...................... Social Networking Service

ToF...................... Time-of-Flight

TUI .... Tangible User Interfaces

Ubicomp .... Ubiquitous computing, a concept in software engineering and computer science where computing is made to appear everywhere and anywhere

UML .... Unified Modelling Language

UWB .... Ultra-Wideband

UX .... User Experience

VNC .... Virtual Network Computing

VPN .... Virtual Private Network

VST .... Video See-Through

WA .... Web Apps

WIMP .... Windows, Icons, Menus, Pointer

WLAN .... Wireless Local Area Network

WORA .... Write Once, Run Anywhere

WPAN .... Wireless personal area network (WPAN)

WM .... Working Memory's Information and Communication Systems

1 Introduction

D2.1, Technology Monitoring: Report on information needed for the Industrial Challenges workers with taxonomy is part of the work in progress of "FACToríeS for WORKERS" (FACTS4WORKERS).

Implement Industry 4.0 worker-centric solutions

The aim of FACTS4WORKERS is to demonstrate the **possibilities of implementing Industry 4.0 worker-centric solutions and increasing shop-floor workers (knowledge) satisfaction** as well as their **productivity in the Factories of the Future**. The project proposal seeks to achieve this objective by defining a set of measureable indicators and limiting the project scope to a set of industrial challenges (use cases). These generalisable industrial challenges are: personalised augmented operator (IC1), worker-centric rich-media knowledge sharing/management (IC2), self-learning manufacturing workplaces (IC3) and in-situ mobile learning in the production (IC4).

Objectives of FACTS4SWOR-KERS

In terms of measureable indicators, the objectives of FACTS4WORKERS are to **increase workers' problem-solving and innovation skills, the cognitive job satisfaction** of workers participating in the pilots and **the average worker productivity by 10%** for workers participating in pilots, and to **achieve TRL 5-7 on a number of worker-centric solutions** through which workers become the smart element in smart factories.

Integrate IT enablers that are already available into a seamless and flexible Smart Factory infrastructure

Smart Factories or Factories of the Future will have **pervasive, networked information and communication technology (ICT)** that collects processes and presents large amounts of data (inventory, machine parameters etc.). Within these factories, workers would require intelligent support from all the modern means that ICT offers. To this end, FACTS4WORKERS will integrate already available IT enablers into a seamless and flexible smart factory infrastructure based on worker-centric and data-driven technology building blocks. We see a Smart Factory as the transformation of central Internet of Thing paradigms to factory automation. This is achieved by applying many technologies that have been proved to work well from applications in the consumer world, offices to the shop floor of factories, where human–computer interaction (HCI) technologies play a leading role, as they fill the gap between the real and the virtual world.

Developing the worker-centric HCI building blocks

WP2 (Worker-centric HCI/HMI building blocks) uses a co-creation process that incorporates shop-floor staff to develop worker-centric HCI/HMI building blocks for the smart factory solution. The maximum possible usability, user experience (UX) and technology acceptance characterise this solution. WP2 builds upon the results of the requirements derived from WP1 (worker needs, organisational requirements and industrial challenges) and is interlinked with the constant evaluation results of WP5 (Deployment: Smart Factory Industrial Challenges). WP2 provides interaction interfaces with the (smart) building blocks delivered by WP3 (Worker-Centric Service Building Blocks) and deployed and maintained by WP4 infrastructure (Smart Factory Infrastructure). WP2 will also be closely related to WP6 (Demonstration & Evaluation of Smart Factory Solution) for the evaluation of the project results. At the same time, some of the information, which is gathered via interfaces, is used to evaluate worker satisfaction. The main objectives of WP2 are to **derive suitable, adaptable worker-centric building blocks** to address worker needs and requirements, **define HCI/HMI concepts and services** for worker-centric building blocks within the smart factory and **develop user-centred interactive technologies** (service front end).

WP 2 objectives

Task 2.1 description

As a preliminary work, or "parallel guiding work", task 2.1, **Analysis of technical requirements and technology monitoring** will analyse available and trending technologies (devices, software developments and tools etc.) within the very dynamic field of today's HCI technology (smart glasses, smart textiles etc.). Task 2.1 will also review new (disruptive) HCI paradigms and connect them to already established (and sometimes outdated) HCI paradigms.

Create a vision of the actual and future developments of HMI technologies and paradigms

The present document D2.1 compiles the results of T2.1, focuns in on the tasks two final objectives: Firstly, **to create a vision of the current and future developments of HMI technologies and paradigms** that will allow other WP2 tasks to gain the maximum benefit from existing technologies when implementing HCI building blocks; and secondly, **to support future technologies adaptation** as they become available over the course of the project's execution. D2.1 will also **provide a general evaluation of existing technologies** by considering their applicability on the factories' shop floor while always keeping mind the project objectives and the industrial challenges indicated in the paragraphs above. The evaluation of the technologies will be provided as a reference for evaluation on a TRL-based scale.

The content of D2.1 has been selected according to the methodology introduced in Chapter 2. Within the restrictions of the project's objectives and scope, we aim to introduce and describe the technology as comprehensively as possible in order to obtain a view that is as general as possible. Because technology is a very dynamic field, relevant technologies can easily be overlooked, which makes this study even more important. Future versions of this deliverable will include updates from academia and commercial developments, as well as evaluations of use case implementations.

The aim to offer a comprehensive result is reflected in the size of D2.1, which was also one of the main issues discussed during this first public release. The participating partners involved in executing WP 2 committed to:

- The provision of technology-related information, which consists of a plain list of technologies and includes a brief summary of each technology in order to ease understanding among a non-expert group of readers.
- Further descriptions and information accessible via references.
- Detailed explanations in the annexes, either in this document or as external documents (for example, online chart sheets).

Figure 1: FACTS4WORKERS will improve working conditions life by using Smart Technologies

2 Methodology for Selecting Technologies and Techniques

This chapter explains the methodology used to determine the technologies reviewed and the document's structure.

Considering Project Goals

The main objective is to meet the aims of the FACTS4WORKERS project and to comply with the projects' restrictions, both inside WP 2 and in relation to other WPs. Excerpting from the project proposal, we can summarise this as an evaluation of how it is possible to use state-of-the-art HCI/HMI technologies to empower workers on the shop floor with smart factory ICT infrastructure.

Center on HCI

This goal is represented on [219] and summarised in Figure 2. This figure presents a simplified overview of a worker in a smart factory environment. The clear separation of the human–machine interaction (HMI) concept and the human–computer interaction (HCI) concept provides an important glimpse of this picture. These two concepts are usually considered synonymous and used interchangeably (see Chapter 3). By differentiating between them, it will be easier and faster to achieve the vision of Industry 4.0 at existing plants.

HCI enables ubiquitous access to Smart Factories information but also supports the inclusion of current machines in the Industry 4.0 vision

Most of the machines already provide HMI interfaces that, among others, make it possible to customise the requirements for part that is being produced or determine the reason for a defect. These interfaces are incorporated into the machine; they are the machine vendor's property, and they are created specifically to convey information between the operator and the machine at a particular moment. Consequently, such interfaces usually do not support the provision of other information, such as manuals. Because these interfaces are attached to a machine, workers physically have to move to the location in order to interact with the machine. Most of the time, this move hampers the execution of their assigned tasks. Another problem is the communication of information from the machine to the Smart Factory Cloud: In many cases, because of their age, it is not possible to connect machines to a network. Sometimes it is not even possible to export certain parameters or basic production data. Fortunately, several such enterprises are around today, including [220], and they can connect external sensors to existing machines and transfer data – in close to real time – to cloud services. From there, the data can be used to provide information visualisation to the workers' mobile HCI. The research in this document is restricted to such HCI technologies.

The lifetime of production machinery can be more than 15 years. This fact, along with the insight that most old machines cannot be connected to a network, leads us to choose the strategy of connecting external sensors to machines and making them network-ready. This strategy will contribute a great deal towards accelerating the implementation of the Industry 4.0 vision, because it will facilitate the co-existence of old and new machines (provided they have standardised hardware/software interfaces and are able to connect to IP networks, among others) and lead to the success of FACTS4WORKERS. Moreover, the use of (adapted) consumer HMI technologies to implement the Internet of Things (IoT) paradigm will assist in taking advantage of most of the latter's features.

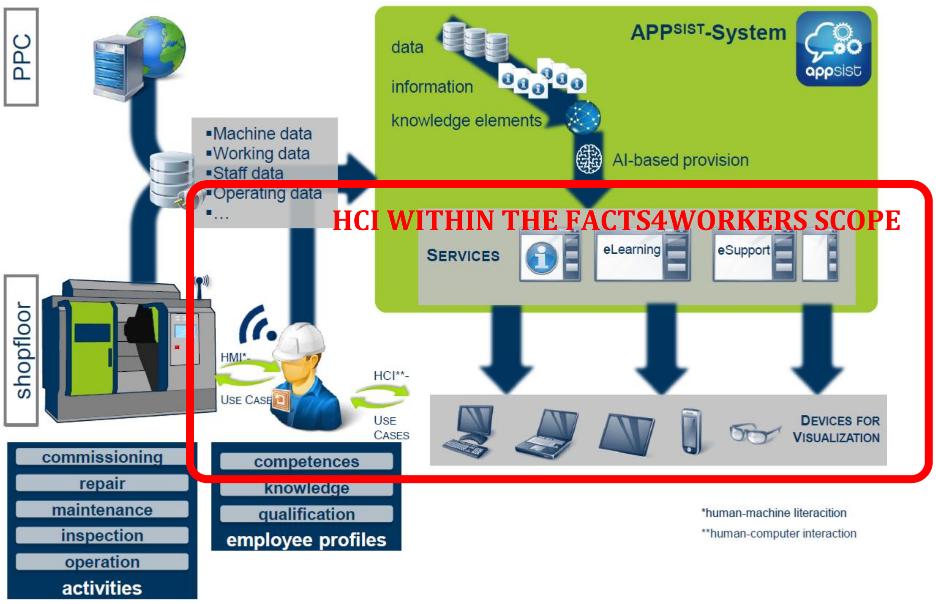


Figure 2: HCI within the scope of FACTS4WORKERS

HCI Smart Factories Solutions should consider HCI IoT advances to make technology transparent

Figure 3, adapted from the one provided in [219], depicts several aspects of a worker's environment: From the manufacturing perspective, Factory of the Future, Smart Factories and Industry 4.0 can be considered the vision of the IoT [221]. By studying the HCI needs of the IoT, we will be able to determine (most of) the technologies to include in this document. An interesting overview of IoT demands is provided in [222]. The authors highlighted that for the IoT vision to be successful, the computing paradigm will need to go beyond traditional mobile computing scenarios that use smart phones and portable devices and evolve to connect everyday objects and embed intelligence in our environment. For technology to move from the user's consciousness to his or her sub-consciousness, the IoT needs to adhere to the following: (1) a shared understanding of the user's situation and appliances, (2) software architectures and pervasive communication networks that will process and convey the contextual information to where it is relevant, and (3) the analytics tools in the IoT that seek to secure autonomous and smart behaviour. The authors based their conclusions on reviewing the concepts of calm computing

[103], human-centric ubicomp [223] and ubicomp in general [224]. It was decided to review the theoretical background of HCI and its individual components (Human, Machine and Interaction) to identify the bases of the technologies currently available as well as those that are still under development in laboratories.

The authors have:

- Reviewed project FACTS4WORKERS objectives and industrial challenges;
- Considered other Industry 4.0 project approaches to solve interaction issues;
- Defined HCI in the Internet of Things vision as Industry 4.0, smart factories and factory of the future as visions of IoT from the manufacturing perspective;
- Conducted profound research on theoretical background.

There are two further steps that were not considered during the first release of D2.1 but form an integral part of this document:

- Combine the obtained taxonomy (see Chapter 6) and implement the use-case prototypes in order to find possible improvements.
- Use the experience of people involved in WP 2 development to review D2.1 content and ensure a comprehensive, understandable, interesting and pragmatic read for both technical and non-technical audiences.

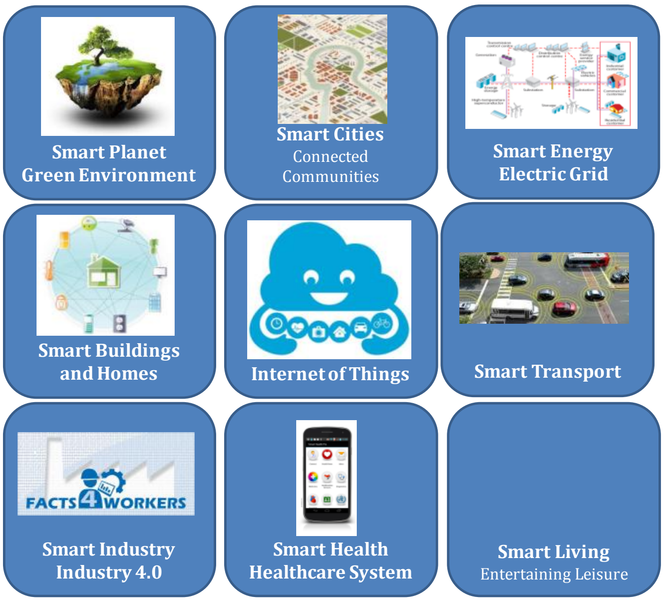


Figure 3: Smart industry in the Internet of Things

After obtaining a clear understanding of the desired technologies (software and hardware) that are to be included in this document, the hierarchy of selection criteria was put into the following order:

Criteria for Building HCI Taxonomy

- Firstly, we grouped them together under HCI-enabling technologies and HCI systems. This group includes individual technologies that others can embed, such as HCI systems, as well as technologies such as touch pads, touch screens etc. A tablet or an augmented reality application is considered an HCI system, because it involves several technologies, e.g. identification/location and/or visualisation technologies.
- The conventional technologies group consists of HCI-enabling technologies, which include devices and technologies that can be classified as state-of-the-art (e.g. keyboard, mouse etc.). The inclusion of these technologies in the document has been actively discussed and was ultimately affirmed for the purpose of being comprehensive. These technologies are also useful when it comes to developing shop floor solutions. For example, T9 keyboards can be used to type short text messages while someone is wearing gloves.
  Although they are not grouped, the rest of the technologies presented under HCI-enabling technologies can be used by themselves (for example, computer vision or context awareness technologies) but are usually included in other more complex technologies (systems). Moreover, they are not as well-known as conventional technologies.
- The HCI System chapter introduces the available technologies that offer an advance in ubiquitous computing, IoT or Industry 4.0 vision from the HCI perspective. In this chapter, we analyse mobile devices, wearable devices and augmented reality as whole systems.

In order to support future versions of this document and relate them to D2.2 and WP2, we have created indexes for chapters 4 and 5 indexes, a taxonomy of technologies for implementation in Industry 4.0 and HCI requirements. This taxonomy, which is presented in Chapter 6, measures the selected technologies against their industry readiness and results in an evaluation of their TRL level, along with the advantages and disadvantages they present. Finally, Chapter 7 presents our conclusions.

Figure 4: Wearables will support the transformation of old machines in Smart Factories

3 Human–Computer Interaction: Theoretical Background

HCI definition and alternative or closely related terms

Human–Computer Interaction (HCI) is a multi-disciplinary area of research and practice that emerged in the 1980s. It focuses on interaction modalities between humans and computers [6] and is defined as "a discipline concerned with ***the design, evaluation and implementation*** of interactive computing systems ***for human use*** and with the ***study of major phenomena surrounding them***" [5]. An important facet of HCI is the ***securing of user satisfaction (End User Computing Satisfaction)***. "Because human–computer interaction studies a human and a machine in communication, it draws from ***supporting knowledge*** on both the machine and the human side."

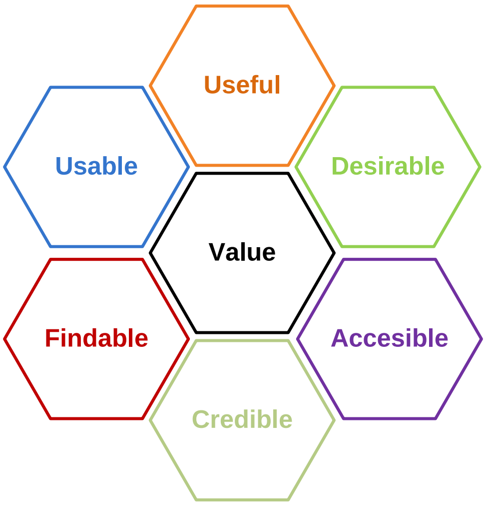


Figure 5: User experience honeycomb

Alternative names for HCI are **Computer–Human Interaction (CHI), Man–Machine Interaction (MMI) and Human–Machine Interaction or Interfacing (HMI**, which is sometimes **used to refer to the user interface in a manufacturing or process-control system**). The term HCI appeared with the emergence of computers. The reason is obvious: For now, even most sophisticated machines are depending on people who will use them properly. Of course, the emergence of so called intelligent machines forsees a reduction in human interaction, but the basis of the machines performance still depends on people who carry out the set-up, programming... etc. These arguments point to the main terms that should be considered during the **design of the HCI: functionality and usability [7]**.

An important concept connected to usability [6] is **user experience (UX)**. It focuses on parameters that relate to the user: **satisfaction**, enjoyability, emotional fulfilment, aesthetic appeal etc. [10]. Research into specialised areas, such as Web interfaces, provides tools that include the UX Honeycomb [9] (see Figure 5), which identifies priorities in the design phase by balancing seven hexagons representing parameters to provide users with a satisfactory **quality of experience (QoE),** which means it will be useful, usable, desirable, findable, accessible, credible and valuable.

To improve the user's quality of experience (satisfaction), the HCI design is based on functionality and usability. It is also important to understand users' mental models.

Another important issue in HCI is how it understands **people's mental models**. Users learn and retain knowledge and skills in different ways that are often influenced by their age and their cultural and social backgrounds. Thus, HCI studies aim to bridge gaps between users and new technologies. **Efficient, effective and natural forms of HCI can reduce the skill levels needed to use complex devices** and leads to a decline in inequality among people by helping to address an issue in the "digital divide".

Table 1: HCI paradigms summary

	Paradigm 1	Paradigm 2	Paradigm 3
Metaphor of Interaction	Interaction as man–machine coupling	Interaction as information communication (human–machine symmetry)	Interaction as phenomenologically situated
Central Goal for Interaction	Optimisation of the fit between man and machine (ergonomics)	Optimisation of the accuracy and efficiency of information transfer	Support for situated action in the world
Typical Questions of Interest	How can we fix specific problems that arise because of interaction?	What are the mismatches that come up during communication between computers and people? How can we accurately model what people do? How can we improve the efficiency of computer use?	What are the existing situated activities in the world that we should support? How do users appropriate technologies, and how can we support those appropriations? How can we support interaction without constraining it too strongly by a computer's ability to do or understand? What are the politics and values at the site of interaction, and how can we support them during the design phase?

Three paradigms explain the HCI evolution: human factors, cognitive science, phenomenological matrix.

Kuhn's theory of the structure of scientific revolutions [3] introduces the term **paradigm** as a way to describe successive and overlapping waves of research that reframe ideas. This theory, along with Agre's theory of generative metaphors in technical work [234], is used to explain the historical evolution of the HCI paradigm. While the former claims that **a paradigm shift is accompanied by a shift in the examples** considered to be central to the field of study (in our case, HCI), the latter suggests that HCI paradigm shifts can be detected by **tracing shifts in the underlying metaphor of interaction**. By applying these theories, three HCI paradigms have been identified: **the human factors (man–machine), the cognitive science and the phenomenological matrix paradigms.** Table 1 introduces them, and they are briefly compared with each other in next paragraphs. Please see Annex A for further details.

A paradigm shift is produced by the movement of the centre of attention to issues that the previous paradigm did not consider, because the technology could not answer these questions, and a new vision would be created. For example, the necessity of a third paradigm was introduced because the second one, focused on information processing, did not support: **ubiquitous computing** (see Chapter 3.1); **the centrality of social, situated actions in explaining the meaning of interaction;** the necessity of new metrics for evaluating user satisfaction, as even performance was are not completely covered by K-12 learning goals; the difficulty to measure efficiency as defined by the first and second paragdigms for non-task-oriented computing; and the marginalisation of emotion.

Table 2: Epistemological distinctions between the paradigms

	Paradigm 1	Paradigm 2	Paradigm 3
Appropriate Disciplines of Interaction	Engineering, Programming, Ergonomics	Laboratory and theoretical behavioural science	Ethnography, action research, practice-based research, interaction analysis
Kind of Methods Strived for	Cool hacks	Verified design and evaluation methods that can be applied regardless of context	A palette of situated design and evaluation strategies
Legitimate Kinds of Knowledge	Pragmatic objective details	Objective statements with general applicability	Thick description, stakeholder "care abouts"
How You Know Something is True	You try it, and it works	You refute the idea that the difference between experimental conditions is because of chance	You argue about the relationship between your data(s) and what you seek to understand
Values	Reduction in errors Ad hoc is OK Cool hacks desirable	Optimisation Generalisability wherever possible Principled evaluation is *a priori* better than ad hoc, since design can be structured to reflect paradigm Structured design better than unstructured Reduction of ambiguity Top-down view of knowledge	Construction of meaning is intrinsic to interaction activity What happens around systems is more interesting than what is happening at the interface "Zensign" – what you don´t built is as important as what you do build Goal is to grapple with the full complexity around the system

The "***phenomenological matrix paradigm***" [2], the third HCI paradigm, was introduced to solve the issues in the paragraph above. It focuses on **the embodiment of interaction: the way in which we come to understand the world and ourselves. Interaction derives from our location in a physical and social world as embodied actors**. Furthermore, thinking is not just cognitive, abstract and information-based but also **achieved through action.** It refocuses attention away from the single-user/single-computer paradigm towards **collaboration and communication through physically shared objects**. Finally, it shows that real-world practice is complex and rich, interweaving physical activity and awareness with abstract thoughts, rituals and social interaction in ways that defy a purely informational approach. For this paradigm, **meaning is constructed** on the fly, often collaboratively, by people in specific contexts and situations, and therefore interaction itself is an essential element in the construction of meaning.

The third HCI paradigm advances in implementing context as base of the interaction.

While the first and second paradigms **acknowledge context primarily as one of "those non-technological factors that affect the use of the technology",** the third considers the context to be a component that is central not only to the problem but also to its design and evaluation. Table 2 compares the three paradigms with each other. The first and second paradigms emphasise the importance of objective knowledge, while the third sees knowledge arising from situated viewpoints. Because of the dominant focus of objective knowledge it is suspected that the complexities of multiple perspectives at the scene of action are not considered in the first and second paradigm.

How to measure success, and recognise innovation are challenges of the third paradigm.

The third paradigm must overcome some **challenges** in order to prove its validity: **measuring success**; **recognising innovation**; have an equivalency of design etc. **Designing interaction moves from attempting to establish one correct understanding set of metrics of interaction to studying the local, situated practices of users and taking into account (but not adjudicating) the varying and perhaps conflicting perspectives of users. Value-based approaches to HCI** (such as participatory design and value-sensitive design) **have come into use to establish new criteria of success – and therefore of decision making – in system design and evaluation [109]**.

3.1 Interaction

The definition of interaction needs to evolve to get technology to serve interaction instead of having interaction conform to technology

Interaction [4] entails the exchange of information between entities (human or machine). Events affecting any of the five human senses are the first part of a form of communication that allows the individual to process, analyse and subsequently act on information about the surroundings and the task at hand.

Figure 6 illustrates a classical human factors interpretation of the human–machine interface [15]. The model cuts down the human and the machine to three components each. The internal state of each factor interacts in a closed-loop system through controls and displays (the machine interface) and motor-sensory behaviour (the human interface). Interaction takes place at the interface between the output channel (e.g. a display), which stimulates human senses, and the input channel.

Two broad themes in this deliverable are interaction and technology. Interaction is the key to pushing the frontier with the aim of improving technology to serve interaction, rather than have interaction conform to technology. As human factors researchers or human–machine system designers, we should fully expect to mould interaction scenarios within a given resource on the basis of today's technology. Although technological constraints tend to vanish simply by waiting for new advances, interaction problems persist, because their solutions require multi-disciplinary research and design efforts that are often ill-defined and qualitative.

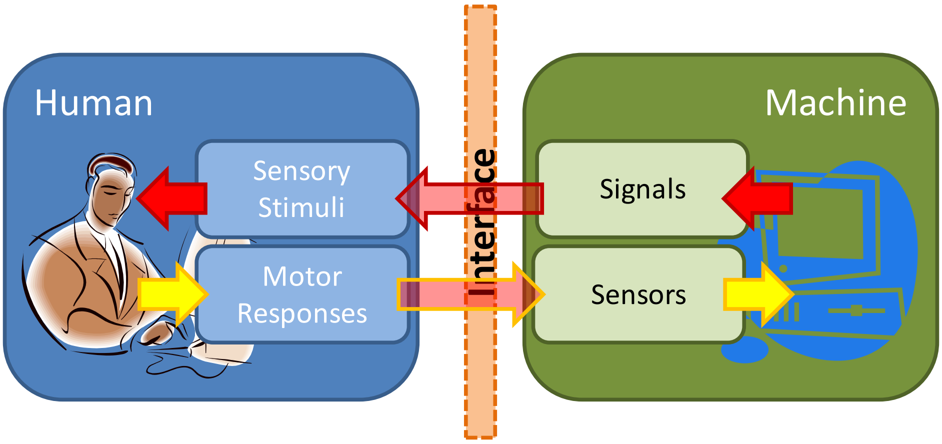


Figure 6: The human–machine interface

As designers, we should look at the **interaction tasks and low-level, primitive inputs required from the user** that are necessary for a particular application. For each task, the designer chooses an appropriate interaction device and **interaction technique** as a way of using a physical device to perform an interaction task.

Interaction paradigms can be identified by applying 5W+H questions to systems

An **interaction paradigm** is a model of HCI that includes all the aspects of interaction. We can analyse systems by applying the 5W+H questions: What/How, Where/When and Who/Why. By asking these questions, we found several different interaction paradigms that are introduced in Table 3: Interaction paradigms according to their relevance to the FACTSW4WORKERS goals (see Annex B for more details).

Technologies advance to the Ubicomp vision of making machine interfaces transparent to users, but the problems that already exist in previous paradigms first need to be resolved.

Table 3: Interaction paradigms

Interaction Paradigm	Sort Description
Large-Scale/ Mainframe Computing	Batch files without interaction (wait till the end). Multi-programming, sharing systems, time-sharing, terminal/text-based interaction, command-line interaction.
Personal Computing/ WIMP	Single user engaged in a dialogue with the computer in order to carry out a series of tasks. Command-line base interaction. WIMP ("windows, icons, menus, pointer") interfaces: Each window runs an isolated programme (multi-programme OS). The main improvements in HCI came about thanks to emulating real-world interactions and providing better ease of use for non-technical personnel. Users can transfer skills in a standardised interface from one application to another.
(Virtual) Network Computing	Network computing seeks to give users access to centralised resources (servers) from simple and inexpensive devices (clients). **Virtual network computing (VNC)**: Servers supply not only applications and data but also an entire desktop environment that can be accessed from any Internet-connected machine by using simple software, such as a thin client. Both approaches are a natural evolution of WIMP interfaces.
Mobile Computing	Mobile computing is the use of transportable computing devices with mobile communication technologies for the transmission of data, voice and video. ISACA mobile computing devices classification: smart phones; laptops; tablets; portable digital assistants (PDAs); portable USB storage devices; radio and mobile frequency identification devices (RFIDs); infrared-enabled devices (IrDAs). Limitations of mobile computing [95][96]: Mobile elements are resource-poor compared with static elements; mobility is inherently hazardous (security problems); mobile connectivity is highly volatile (performance and reliability); and mobiles rely on a finite energy source.
Wearable Computing	Wearable computers are miniature electronic devices that the wearer carries under, with or on top of clothing. Features: consistency (there is constant interaction between computer and user) and user multi-tasking (not necessary to stop what you are doing to use the device). Common issues with mobile computing, ambient intelligence and ubiquitous computing research communities: power management, heat dissipation, software architectures and wireless and personal area networks.
Collaborative Computing	Collaborative computing [97] uses computers to support coordination and cooperation of two or more people who attempt to perform a task or solve a problem together [111]. Systems supporting it are referred to as groupware.
Virtual Reality	Virtual reality (VR) is an immersive multimedia or computer-simulated life. It replicates an environment that simulates physical presence in places in the real world or imagined worlds and lets the user interact in that world. Sensory experiences can include sight, hearing, touch, smell and taste. When VR covers remote communication environments, it provides the virtual presence of users with the concepts of telepresence and telexistence or a virtual artefact (VA) either through standard input devices, such as a keyboard and mouse, or through multimodal devices, such as a wired glove or omnidirectional treadmills.

Augmented Reality	Azuma [72] provides a commonly accepted definition of AR as a technology that (1) combines real and virtual imagery, (2) is interactive in real time and (3) registers the virtual imagery with the real world. See Chapter 5.5.
Natural Interaction	Natural interaction is based on a natural user interface (NUI). This is a system for HCI that the user operates by means of intuitive actions related to natural, everyday human behaviour. Some examples and applications of natural user interfaces: touchscreen interfaces (see Chapter 4.2); gesture recognition systems (Chapter 4.5); speech recognition (Chapter 4.7.1); gaze tracking (Chapter 4.1.3) or brain–machine interfaces.
Multimodal Interaction	Multimodal interaction [105] refers to interaction with the virtual and physical environment through natural modes of communication and involves the five human senses. Multimodal systems offer a flexible, efficient and usable environment allowing users to interact through input modalities (i.e. speech, handwriting, hand gesture or gaze) and to receive information from the system through output modalities (speech synthesis, smart graphics etc.). On the one hand, the advantage of multiple input modalities increases usability, which is important for solving accessibility problems, in particular for people who are [117] “situationally impaired”, for example if they are wearing gloves. On the other hand, a multimodal output increases synergy and redundancy as well as the bandwidth of information transfer [118]. Finally, it would facilitate an invisible interface space using sensor technology (infrared, ultrasound and cameras) [119].
Adaptive Interfaces	A user-adaptive[107] system is an interactive system that adapts its behaviour to individual users on the basis of processes of user model acquisition and application that involve some form of learning, inference or decision making. The main functions are to: support system use; take over parts of routine tasks; adapt the user interface so that it fits better with the user’s way of working with the system; help with system use; mediate interaction with the real world; control the dialogue it maintains with the user; support information acquisition: help users to find information; recommend products; tailor information presentation; support collaboration; and support learning.
Ubiquitous/ Pervasive Computing	Ubiquitous computing is the method of enhancing computer use by allowing for many computers to be available throughout the physical environment but rendering them effectively invisible to the user [103]. Ubicomp attempts to break away from the paradigm of desktop computing to provide computational services to a user when and where required. Rather than force the user to seek out and find the computer's interface, ubiquitous computing suggests that the interface itself can take on the responsibility of locating and serving the user. Ubicomp systems are concerned not only with software services but also with devices and how to combine them. According to Weiser [103], Ubicomp has two main attributes: ubiquity interaction, meaning the system is available wherever the user needs it (user mobility); and transparency, in other words, the system is non-intrusive and integrated into the everyday environment. A ubiquitous computing system can be also characterised by the provision of two services: context awareness (applications should adapt themselves based on knowledge of location); and automated capture, integration and access (use computational resources to augment the inefficiency of human record taking, to automate explicit and implicit links between related but separately generated streams of information and to support access that would aid in recalling the meaning or significance of past events).

3.2 Human

Models like MHP represent human complexity in HCI

From the HCI perspective, human complexity has to be represented by models that help to understand human behaviour, reactions, etc. The most accepted model is the **model human processor (MHP),** which Card [19] introduced. It proposes a high-level description of the human brain that helps to understand and predict user–computer interaction. It is based on the high-level description of an information-processing system in terms of memories, processors, parameters and interconnections. This basis helps to envision the system as a whole and to make predictions of total system behaviour. Following the same approach, the human mind can roughly be described as an information system. The intention of the model is not to represent precisely what is inside the worker's mind but to help to remember facts and to make predictions of human behaviour.

The MHP comprises three subsystems: **the perceptual, cognitive system and motor systems**. Each subsystem has a processor (characterised by its cycle time) and memory (defined by its capacity and decay time) [20]. Figure 7: MHP model represented by Baley [19] presents a high-level view of MHP. Although this model does not include the haptic sensory processor and memory, it can easily be extended to include them.

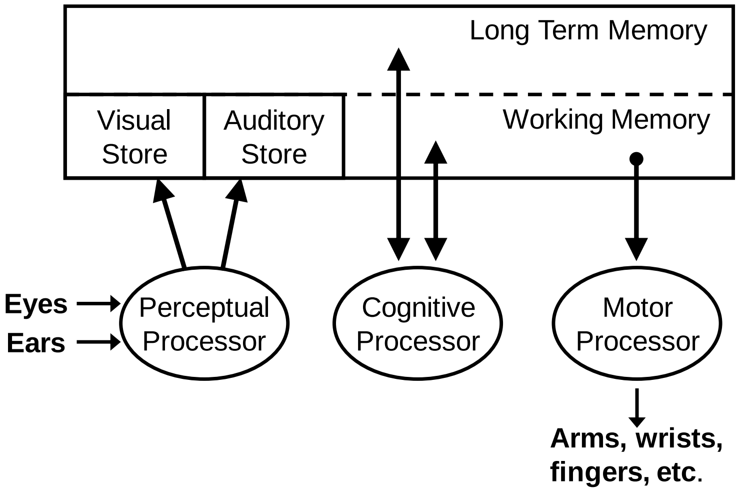


Figure 7: MHP model represented by Baley [19]

MHP principles of operation explain different responses in different contexts

MHP also defines a number of **principles of operation** dictating the behaviour of the system under certain conditions, in other words according to the context. Sometimes systems operate in serial (a key is pressed in response to a stimulus), while at other times they operate in parallel (driving and talking to a passenger or listening to the radio). A more detailed presentation of the model and its relations can be found in Annex B.

During an interaction with a computer, the human input is the computer data output and vice versa. Input in humans occurs mainly through the senses (perceptual subsystem) and output through the motor controls of the effectors (motor subsystem). Vision, hearing and touch are the most important senses in HCI. The fingers, voice, eyes, head and body position are the primary effectors [22]. Annex D contains a more detailed explanation of each of the human "*input–output channels*" as well as of the motivators and constraints that influence perception and/or actuation.

The main differences (for the moment) between a human and a machine are that humans have emotions and are able to think (in other words, to reason and consequently solve problems). The cognitive subsystem performs these functions.

Humans are able to think (to reason and to solve problems) and to express emotions, whereas computers are not, at least not yet

Thinking requires different kinds and amounts of knowledge. For the sake of simplicity, we will divide it into reasoning and problem solving. **Reasoning** is the process by which we use the knowledge we have to draw conclusions or infer something new about the domain of interest. **Problem solving** is the process of finding a solution to an unfamiliar situation by using (adapting) the knowledge we have.

Emotions concerns both physical and cognitive events. Our body responds biologically to an external stimulus, which we interpret as a particular emotion. This biological response (affect) changes the way we deal with different situations and has an impact on the way we interact with computer systems.

3.3 Computer

As we introduced in Chapter 3.1, interaction (with or without a computer) is the process of information transfer. The diversity of devices reflects the fact that there are many different types of data that may be entered into and obtained from a system and many different types of users [21]. Traditionally the computing literature has often made a sharp distinction between input and output; computer scientists are used to regarding a screen as a passive output device and a mouse as a pure input device [13]. However, nearly all examples of human–computer interaction require both an input and an output to function as a useful system. For example: What good would a mouse be without the corresponding feedback embodied by the cursor on the screen, as well as the sound and feel of the buttons when they are clicked?

The differentiation between input and output devices has to be based on the function for the user

Following on from the previous point, a good way to classify input/output devices is to use the function as perceived by the user, instead of the direction of the computer data flow, as a classification criterion. This idea guides the work of Dix and colleagues [21], and we use it as a short introduction to the interaction devices explained in the next paragraphs.

Input devices move information from the user's brain to the computer

Input to computers consists of sensed information about the physical environment [13]. In the context of HCI, the fundamental task in computer input is to move information from the brain of the user to the computer. Traditionally, human outputs are our limbs – the hands, arms, legs, feet or head – but speech and eye motions can also act as human input. Some human output channels are breath and electrical body signals (important for disabled users). Work in this area attempts to increase the bandwidth across interfaces by seeking faster, more natural and more convenient means for a user to transmit information to a computer.

For example, user performance with many types of manual input depends on the speed with which the user can move his or her hand to a target. Fitts's Law (see Annex E) provides a way to predict this and is a key foundation of input design. It predicts the time required to move on the basis of the distance to be moved and the size of the destination target.

Output devices transfer the results of an information-processing system to the human brain

An **output device** is any piece of computer hardware used to communicate the results (data) of an information-processing system (such as a computer) that converts electronically generated information into human-readable form [122]. Output devices can be classified as text, graphics, tactile [124], audio and video [123].

Text consists of characters (letters, numbers, punctuation marks or any other symbol requiring one byte of computer storage space) used to create words, sentences and paragraphs. Graphics are digital representations of non-text information such as drawings, charts, photographs and animation (a series of still images in rapid succession that gives the illusion of motion).

Tactile output, such as raised line drawings, may be useful for individuals who are blind. More generally haptic technology, or haptics, is a tactile feedback technology that takes advantage of the sense of touch by applying force, vibration or motion to the user. Several printers and wax jet printers have the capability to produce raised line drawings. There are also handheld devices that use an array of vibrating pins to present a tactile outline of the characters or text under the viewing window of the device.

Audio is music, speech or any other sound. Speech output systems can be used to read screen text to computer users. Special software programs called screen readers attempt to identify and interpret what is being displayed on the screen, and speech synthesisers convert data to vocalised sounds or text.

Videos consist of images played back at speeds to provide the appearance of full motion.

Figure 8: Smart factories require the co-existence of disruptive and well-established interaction technologies

4 HCI-Enabling Technologies

HCI-enabling technologies are embedded in bigger systems

This chapter briefly introduces what we consider as HCI-enabling technologies: technologies that, in most cases, are self-contained subsystems and usually embedded in other systems, such as a computer, a smart phone or a smart watch. We first look at conventional technologies, which we assume are known to almost everyone and include keyboards, mice etc. Subsequently, we will introduce more modern and (sometimes) innovative technologies, at least with regard to the industrial shop floor.

We differentiate between enabling and complex technologies, because we think it is a way to identify some of the building blocks to be created within the WP2 of FACTS4WORKERS and as a first step to create the taxonomy of HCI technologies for the shop floor. For example, by considering computer vision as an enabling technology, we can classify the construction of a building block that implements the desired functionalities (OCR, object recognition etc.) as a subsystem that can "easily" be ported into different, more complex systems (a smart phone, PC etc.) in order to achieve certain functionalities or implement different use cases.

4.1 Conventional Technologies

The WIMP interaction paradigm implements conventional technologies

The concept of conventional technologies includes those technologies and devices that, to some extent, are expected to be included in a PC from the 1980s. A conventional technologies HCI is mostly based on the visualisation of data through a screen, and information is entered using a keyboard and sometimes a mouse, in line with the WIMP interaction paradigm (see Chapter 3.1).

Including or not including such technologies in this document has been the object of internal consortium discussion. The conclusion was that even though they have already been implemented in many devices, it is important to keep them as part of our analysis. Firstly, we keep them in order to highlight the capabilities of new options in comparison with old functionalities (voice recognition instead of a touchpad or a keyboard to introduce commands). Secondly, by keeping them, we avoid forgetting old solutions when trying to solve a future problem (for example, T9 keyboards can support some text entering when gloves are used, because keys can be bigger).

4.1.1 Text Entry

Text entry is the most common and the oldest way of HCI. A text entry interface or text entry device is an interface that is used to enter text information into an electronic device. From punch card input to speech recognition technology, it has undergone a remarkable evolution. Nevertheless, mechanical computer keyboards are still the most commonly used (input) device.

Text entry is the most common way to enter information into systems

Keyboards

Keyboards are the main input devices in use today. There are various types of keyboards, and different features distinguish them from each other. Firstly, there is a distinction to be made as to whether they are physical, like the keyboards attached to PCs, or embedded in laptops; or virtual, like the ones provided by smart phones and tablets. Virtual keyboards can also be classified as visual-touch keyboards (on mobile devices like smart phones) or projected-vision detection (see chapters 4.3 and 4.4), which combines projector capabilities with motion detection and object recognition.

With physical devices, there are some important features to consider. Size is probably the most important, because on this basis the keyboard could be extended, including the numerical keyboard as well as other function keys (allowing the execution of common complex commands without the need to use a combination of keys). The keyboard size will also define the size of the keys as well as the spaces between them; for some industrial uses, this point would determine if it is possible to operate the keys while wearing gloves. Ease of use would also depend on whether the keyboard is suitable for rugged use in a rugged environment. Finally, according to the HCI aims to improve ergonomics, some of the available keyboards in the market are designed to improve typing performance as well as to protect user health. Split keyboards, angled split keyboards, contoured keyboards and handheld keyboards are some examples of ergonomic implementations.

A common feature of physical and virtual keyboards is their key layouts, which have a major impact on their usability and performance. Table 4: Keyboard layout summary introduces and describes the most significant among them.

Text Entry Innovative Ways

We briefly introduce innovative ways to input text into cyber-physical systems already available in consumer technologies today. These solutions will be discussed in more detail in Chapter 4.1.1.

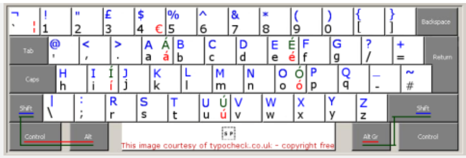
When hands-free is a necessity, state-of-the-art solutions such as speech recognition should be considered

Firstly, we take a closer look at **handwriting recognition (HWR)**[26]. This term refers to the ability of a computer to receive and interpret intelligible handwritten input either offline, from sources such as paper documents or photographs and based on OCR, or online, by sensing the pen tips on a touchscreen or another kind of device. These technologies are explained further in Chapter 4.4.

Secondly, we discuss virtual keyboards that are projected onto a surface (see Chapter 4.3) and combine the projection of the keyboard with detection using computing-vision technologies to recognise the pressed keys.

Finally, another type of text input comes from the user's speech: **speech recognition**. Although its performance is highly dependent on the noise environment as well as the language used, it has performed well for restricted vocabularies and is useful for activities that demand the use of both hands, such as driving or doing a task on the factory shop floor. Its advantages and disadvantages are explained in Chapter 4.7.1.

Table 4: Keyboard layout summary

Name and Description	Picture
The **QWERTY** layout is not optimal for typing but dates from the time of mechanical limitations of the typewriter.	
Alphabetical keyboards arrange keys in **alphabetical order**. While they could be useful for people who do not use QWERTY keyboards, they create confusion for people who use them.	
The **DVORAK** layout minimises the stretch of fingers and the use of weak fingers, reduces fatigue and increases typing speed (10-15%). It is not very common and thus could lead to some confusion.	
Chord(ed) keyboards or keypads are keyboards with a few keys, where a single hand presses combinations of up to five keys to represent different characters. They are smaller than conventional keyboards and have a short learning time. T9 entry, text in nine keys, uses the numeric keys of the keypad on a cell phone. The popularisation of T9 keyboard contributed to the creation of the **T9-Predictive Algorithm** (XT9 Keyboard), whose objective is to make it easier to type text messages. It allows words to be entered by a single key press for each letter, as opposed to the multi-tap approach used in conventional mobile phone text entry, in which several letters are associated with each key, and selecting one letter often requires multiple key presses.	  

4.1.2 Display Devices

A **display device** is an output device for the presentation of information (text, graphics or video) in visual (using a screen) or tactile [125] form (tactile electronic displays for blind people). Information shown on a display device is called **soft copy,** because the information exists electronically and is displayed only for a brief period of time.

Display devices provide a soft copy of information mainly through the visual channel

Common attributes to consider when comparing the quality of visual display devices are:

- **Resolution,** which describes the sharpness and clearness of the image and is directly related to the number of pixels a monitor can display. A greater number of pixels results in a higher-quality image.
- **Dot pitch,** which measures image clarity and calculates the distance between each pixel on a display. A smaller distance between pixels means a sharper image.
- **Refresh rate,** which is the speed at which a monitor needs to redraw images on the screen. The refresh rate should be fast enough to maintain a constant, flicker-free image.
- **Colour gamut** [126] or colour depth. In colour reproduction (including computer graphics), this term refers to a complete subset of colours that can be accurately represented in a given circumstance, such as within a given colour space or by a certain output device.
- **Colour accuracy,** which measures how accurately a device renders a colour.
- **Space ratio,** which represents the ratio of the screen's display width to the height.
- **Screen size,** which measures the distance in inches from a screen corner to the one diagonally across from it.

Screens

Monitors for visualisation, or screens in general, include a range of display devices from obsolete CRT monitors to LCD and more recently OLED devices without any consideration of the system utilising them (television, computer, smart phone etc.), as these uses converge and so do many of their features. A complete list of the different types of devices, based on their supported technologies, is provided in [130], which includes a study on their energy consumption.

Screens are the most common display device, and their power consumption determines where they can be used

Table 5: Visualisation monitor technologies summarises the available technologies based on the additional criteria of ability to be used by smart and wearable devices or by computers today.

Flexible displays are flat computing screen panels constructed out of thin or flexible substrate that can be bent, rolled, folded or flexed without loss of functionality. The flexible substrate used to replace conventional glass can be plastic or thin glass. The displays can be printed or deposited into thin foil.

Table 5: Visualisation monitor technologies

Power Consumption	Technology	Description
Low	e-paper	Electronic paper [131] emulates traditional paper. E-paper belongs to the non-emissive displays but does not require backlight; light from the immediate environment is sufficient. The limits of e-paper include a very low refresh rate, and a shadow of an image may be visible after parts of the screen have been refreshed. By contrast, it has low power consumption, can retain an image without needing a battery and has a wide view angle. It does not provide a complete colour gamut. Applications: electronic pricing labels in retail shops, digital signage, timetables at bus stations, electronic billboards, mobile phone displays and e-readers able to display digital versions of books and e-paper magazines.[132]
Low	OLED	An organic light-emitting diode [134] is an emissive organic material that can produce a full-colours flat-panel display with the help of an electrical current. The chemical composition of the very thin layers of organic material dictates the colours that are produced. OLEDs are so thin that they can be placed on plastic film. They work without a backlight [134]. OLEDs can be used in: gaming and mobile devices, video cameras, computer monitors, laptops and TVs. Recent implementations are AMOLED and super AMOLED. Compared with LCD, OLED provides: lower cost; light weight and flexible plastic substrates (such as roll-up displays embedded in fabrics or clothing); wider viewing angles and improved brightness; better power efficiency and thickness; better response time; thinner and lighter screens; higher contrast within low ambient light conditions. Some disadvantages include: a shorter lifespan; worse colour balance; less efficiency of blue OLEDs; water damage; worse outdoor performance; up to 40% higher power consumption.
Low	LCD	Liquid crystal display technology works by blocking light. An LCD is made of two pieces of polarized glass that contain a liquid crystal material between them. A backlight creates light that passes through the first substrate. At the same time, electrical currents cause the liquid crystal molecules to align to allow varying levels of light to pass through the second substrate and create the colours and images. LCD displays additional attributes to consider include: native resolution (the resolution for which they are designed); viewing angle (maximum angle at which a display can be

		viewed with acceptable visual performance); brightness or luminance (measurement of the amount of light the LCD monitor produces, given in nits or candelas per square meter); contrast ratio (degree of difference of ability to produce bright whites and the dark blacks); response rate (how fast the monitor's pixels can change colours); adjustability (possibility to swivel, tilt up and down and rotate from landscape to portrait mode).
Low	ELD	Electroluminescent displays (ELDs) are a type of flat-panel display created by sandwiching a layer of electroluminescent material between two layers of conductors. When current flows, the layer of material emits radiation in the form of visible light. 

Head-Mounted Devices: Smart Glasses.

Head-mounted displays (HMDs) are small displays or projection technologies integrated into eyeglasses or other devices worn on the head. They are used to provide virtual-reality or augmented-reality experiences (see chapters 5.2.2 and 5.5).

HDMs and HUDs support the development of AR

Heads-up displays (HUDs) are a type of HMD that does not block the user's vision but superimposes an image onto the user's view of the real world. An emerging form of HUD is a display integrated into/or paired with contact lenses. In all cases, the user perceives the virtual image at an ideal viewing distance, even though no screen is present.

Projectors

A **projector**, also known as digital light processor (DLP), is a display device based on optical micro-electro-mechanical technology that uses a digital micro-mirror device. DLP is used in a variety of display applications from traditional static displays to interactive displays and also non-traditional embedded applications, including medical, security, and industrial uses. DLP technology is used in DLP front projectors (standalone projection units, primarily for classrooms and business), DLP rear-projection television sets and digital signs. It is also used in about 85% of digital cinema projectors and as a power source in additive manufacturing in some printers to cure resins into solid 3D objects.

Projectors can also be used to implement AR applications

Within the scope of FACTS4WORKERS, we also look at **handheld projectors, sometimes known as pocket projectors, mobile projectors, pico projectors or mini beamers.** This technology applies an image projector to a handheld device

[135]. It comes in response to the emergence of compact portable devices such as mobile phones, personal digital assistants and digital cameras, which have sufficient storage capacity to handle presentation materials but little space to accommodate a display screen. Handheld projectors involve miniaturised hardware and software that can project digital images onto any nearby viewing surface.

Projector features

The defining characteristics of (handheld) projectors are [137]: resolution and brightness (measured in ANSI lumens; typical pico projectors are under 2,000 lumens) Naturally weigth and cost a part of a projectors selection criteria as well.. Additionally, some other features and functionalities to be considered include: zoom lens (gives you the ability to adjust the projected image size without physically moving the projector); key stone correction (the capacity to correct the effect which is produced when projecting an image from any angle other than straight onto the projection surface, the result is an image that is not completely square, instead appearing trapezoidal); contrast (the ratio between the brightest and darkest areas of the image, less important than lumen output); video signal standards (S-video, RGB, DVI, HDMI etc.).

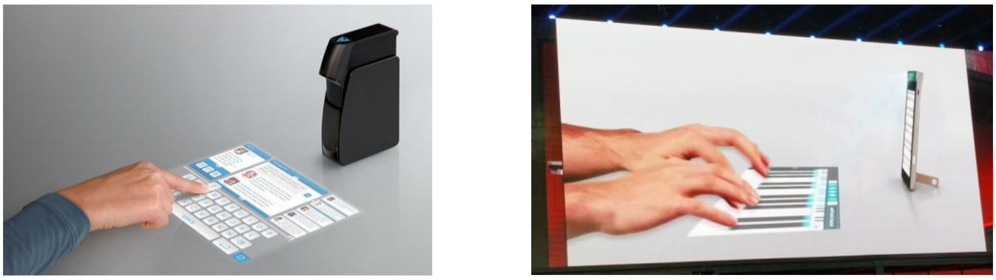


Figure 9: CES 2010 micro projector and Lenovo projector phone

Projectors, when combined with a camera and CV, can be used to recognise gestures

A very interesting feature is the possibility of interacting with the projected images. Interactive projectors create an electronic whiteboard on any surface where the image is projected and allow the user to interact with the projected images.

Finally, it must be noted that although most projector systems are based on “conventional” video technologies, laser projection is also an option [162] .

4.1.3 Screen Positioning, Pointing and Drawing Technologies

Pointing is the base of the WIMP paradigm

Screen positioning, pointing and drawing technologies are used to signal something on a screen. Within the WIMP interaction paradigm (see Chapter 3.1), the technologies involve a screen, a pointer (a symbol in the screen) and a pointing device that is used to control the movement of the pointer on the screen and to perform an action (selecting, clicking etc.). As we all know, this has been simplified by new technologies (such as touchscreens) that make it possible to signal and to actuate without the use of external devices. The next few paragraphs will review more relevant technologies for the purpose of the FACT4WORKERS project scope.

Most devices used for manual pointing or locating can be categorised in the following ways [11]:
Type of motion: linear vs rotary
Absolute or relative measurement
Physical property sensed: position (or angle) or force (torque)
Number of dimensions: one, two or three linear and/or one, two or three angular
Direct vs indirect control: Mouse is indirect (move it on the table to point to a spot on the screen), while touchscreen is direct (touch the desired spot on the screen directly)
Position vs rate control
Integral vs separable dimensions

Classical Pointing Technologies

We first consider devices that we see as classical in the sense that they can already be found on a factory shop floor. Their features, advantages and disadvantage are shown in Table 6: Conventional pointing devices.

Table 6: Conventional pointing devices

Name	Description
Mouse 	A mouse is an indirect input device that detects two-dimensional motion relative to a surface, which is translated into the motion of a pointer on a display allowing fine control of a GUI. It has one or more buttons that can be assigned to different functions. The buttons may also change their function if they actuate in coordination with keyboard special keys. The technology used to gather movements can be mechanical, optical, inertial or gyroscopic in nature. An inertial mouse seems to be the most appropriate to be used in an industrial environment to support industry conditions.
Touchpad 	A touchpad or trackpad [34] is a pointing device featuring a tactile sensor, a specialised surface that can translate the motion and position of a user's fingers to a relative position on the operating system (OS) displayed on the screen. Touchpads are a common feature of laptop computers and are also used to substitute a mouse if desk space is scarce.
Trackball and Thumbwheel 	A trackball is a pointing device consisting of a ball held by a socket containing sensors to detect a rotation of the ball in two axes. The user rolls the ball with the thumb, the fingers or the palm of the hand to move a pointer. Unlike a mouse, a trackball has no limits on effective travel.
Joystick and Keyboard Nipple   	A joystick is an input device consisting of a stick that pivots on a base and reports its angle or direction to the device it is controlling [36]. It often has supplementary switches that are used to access different controls. There are different kinds of joysticks: **absolute sticks**; **isometric sticks**; **analogue sticks** [37] (aka **control stick or thumb stick)**; **keyboard nipples or pointing sticks,** which are tiny joysticks sometimes used on notebook computers. Joysticks are present in most aircrafts today. They are also used in many industrial and manufacturing applications to control machines such as cranes and trucks. Additionally, most unmanned aerial vehicles (UAVs) and submersible remotely operated vehicles (ROVs) require at least one joystick to control the vehicle, on-board cameras, sensors and manipulators. **Miniature finger-operated joysticks** have been adopted as input devices for smaller electronic equipment such as mobile phones and some smart-glasses models.
Stylus and Light Pen 	The light pen [31] is an input device in the form of a light-sensitive wand used in conjunction with a computer's display. It allows the user to point at displayed objects or draw on the screen in a similar way to a touchscreen but with greater positional accuracy. A more modern technology uses a stylus in combination with a tablet. This kind of technology is used for: **pointing/locator input**; **performing handwriting recognition**; **direct manipulation; gesture recognition** (as indicators for commands).

Innovative Pointing Technologies

Touchscreens simplify interaction

We now briefly introduce some new technologies that are used to interact with the virtual world. These technologies, their advantages and disadvantages are explained in more detail later in this document, as pointed out in the relevant sections.

A touchscreen [39] is an electronic visual display (LCD, CRT or projected) that can detect the presence and location of a touch within the display area. Most of our mobile phones are equipped with such screens. They have two main attributes: They enable direct interaction with what is displayed, rather than indirectly like a mouse or touchpad, and the interaction does not require a handheld device. More details can be found in Chapter 4.2.

Gesture recognition and eye tracking are other ways to make progress with natural interaction

A second kind of innovative pointing interaction is based on gesture recognition (GR), in particular hand–arm gestures recognition and head-and-face gestures recognition. Gestures are human high-level events that directly map "user intent" with the desired object and/or action without forcing the user to learn and remember operational details of commands and options. GR is supported by computer vision or sensor-based methods. They share the feature of being hands-free and applicable to unclean environments. They can be used by “situationally impaired” people (see Chapter 3.1), for example workers who are driving or controlling robots. Their features, advantages, disadvantages and challenges can be found in chapters 4.4 and 4.5.

A third group of technologies that can be used to signal and interact with an application is based on eye-tracking algorithms that use computerised vision algorithms to track eye movement while actions are carried out. Eye tracking is very precise if the gaze is fixed, but its quality decreases with moving targets. Eye tracking is presented in detail in Chapter 4.6

Finally, BCI (brain–computer interfaces) are a set of promising technologies that will provide HCI to disabled people in the future (see Chapter 4.10).

4.1.4 Printers

Within the classical conception, a printer [123] is an output device that produces text and graphics on a physical medium such as paper or transparency film, the result being a hard-copy output. This conception has been expanded with the creation of 3D printers, which are able to produce (print) physical representations of virtual objects in three dimensions and are used in additive production processes. Both 2D and 3D printers have different features determining their application fields. Table 7: 2D Printers introduces the different types of 2D printers.

The main features of 3D are presented in Annex F.

Table 7: 2D Printers

2D RFID printers will play an important role in the implementation of AR applications, as they are used to print identifications and locators

Type of Printer	Features
Impact	They form characters and graphics by striking a mechanism against an ink ribbon that physically touches the paper. A dot-matrix printer is an impact printer that prints images when tiny wire pins (7-24) on a print-head mechanism strike an inked ribbon. **Advantages**: They can print on multi-part stationery or make carbon copies; they have low printing cost; they operate under difficult environmental conditions; and they last a long time. **Disadvantages**: noise; low resolution; very limited colour performance; low speed.
Thermal	The many different kinds of thermal printers include ribbon thermal printers, thermo-sensitive paper printers, solid ink printers and dye-sublimation printers. Thermal printers make little noise, their costs are low, and they are compact. The quality of the result depends on the technology and the materials. Some can be used to print colours with cost for each printed copy.
Inkjet Printers	Inkjet printers spray a very small amount of ink onto the media using a piezoelectric element. Their cost is low, their size is compact, and they make little noise. Their colour printing quality is acceptable. Some inkjet printers may require special paper. Usually inkjet printers are slower than laser printers.
Laser Printers	Laser printing is the most advanced 2D printing technology. In laser printing a computer sends data to the printer, which translates these data to printable image data. These kinds of printers use the xerographic principle. A laser beam is shone into a photosensitive drum, which creates a latent image on the drum. During the development process, the toner is attached to the drum surface and then transferred to the paper. The quality of laser printing is high. These printers offer low noise and high speed. They are more expensive than inkjet or dot-matrix printers, and they are generally larger in size.

There are different types of 2D printers. Their classification depends on the kind of material that can be used to print, the technology used to put the text or the image onto the material, the possibility of printing in colour, the resolution (dots per inch), the printing speed and the quality at that speed, as well as the maximum supported printing size and the possibility of having network connectivity.

Special 2D printers are RFID printers: They can “print” RFID labels, which are used to identify products or locations. RFID smart label printers/encoders use media that have an RFID inlay (chip and antenna combination) embedded in the label material. An RFID encoder inside the printer writes data to the tag by means of a radio frequency transmission. The transmission is focused on the specific location of the tag within the label. Bar codes, text and graphics are printed as usual. RFID printers are characterised by the kind of RFID tags that they are able to write as well as by the “visual” printing capabilities they provide.

As with 2D printers, the main considerations in choosing a 3D printing machine tend to be speed, costs of the device and per printed prototype, choice and cost of needed the materials, and colour capabilities.

Figure 10: RX900 colour RFID printer

4.2 Touch-Sensitive Screens (Touchscreens)

Touchscreens are input/output devices supporting direct interaction with displayed images.

A touchscreen [39] is an electronic visual display (see 4.1.2) that can detect the presence and location of a touch within the display area. The term generally refers to touching the display of the device with a finger, hand or pen/stylus. It has two main attributes. Firstly, **it enables one to interact directly with what is displayed** rather than indirectly with a cursor controlled by a mouse or touchpad. Secondly**, it lets one do so without any intermediate device that would need to be held by hand**.

Touchscreens are common in devices such as game consoles, PCs, tablet computers, eBooks and smart devices (phones, watches etc.) [38]. Touchscreens are also found in the medical field **and in heavy industry**, as well as on automated teller machines (ATMs) and at kiosks such as museum displays or room automation. Touchscreens are mostly used **where keyboard and mouse systems do not allow a suitably intuitive, rapid or accurate interaction between the user and the display's content.**

The technology used in the implementation of the touchscreen determines whether it would be usable on the industrial shop floor

The touch panels themselves are based around four basic screen technologies: **resistive touchscreen; capacitive touchscreen; infrared touchscreen; surface acoustic wave (SAW).** Each of these designs has distinct advantages and disadvantages. A brief introduction of each kind is provided in the following paragraph, and a more detailed study of each can be found in the referenced documents. Their features and functionalities are compared with each other in Table 8: Comparison between touchscreen features.

4.3 Image and Video Devices

Most handheld devices today are equipped with at least one digital camera

A camera is an (optical) instrument with which to record images to be stored locally, transmitted to another location, or both. The images may be individual still photographs or sequences of images constituting videos or movies. This section reviews some of the most significant camera technologies that are available on the market and are suitable to support workers in their daily tasks.

At present, most computers, tablets and phones are equipped with one or more digital cameras to take photographs or to record videos. A digital camera is a camera that encodes images and videos digitally and stores them for subsequent reproduction.

Table 8: Comparison between touchscreen features

Properties	Resistive 4/5/8-Wire	Digital	SAW Surface Acoustic Wave	Capacitive Projected	Surface	Infrared
Type Surface	Electrical, analogue	Electrical, digital	Surface technique (acoustic)	Capacitive electrical field through material(s)	Capacitive electrical field on material surface	Edge technique (optical)
Durability	Low	Low	Low	High	Medium	High
Transparency	Bad	Bad	Good	Normal	Good	Good
Stability	High	High	Good	Normal	Good	Good
Touch	Anything	Anything	Finger/pen	Conductive	Conductive	Finger/pen
Response Time	<10ms	<15ms	10ms	<15ms		<20ms
Sensitivity	Very sensitive to scratch	Very sensitive to scratch	Sensitive to scratch	Sensitive to dirt	Sensitive to dirt	Ambient Light
Following Speed	Good	Good	Low	Good	Good	Good
Chemical Resistance	Alcohol, acetone, grease and general house-hold detergent	Acetone, methylene chloride, turpentine, isopropyl alcohol, hexane, tea, vinegar and coffee	Resistant to all chemicals that do not affect glass, including acetone, toluene, gasoline, kerosene and vinegar	Resistant to all chemicals that do not affect glass, including acetone, toluene, gasoline, kerosene and vinegar	Does not work when wet or moist	Alcohol, acetone, grease and general household detergent
Water Proof	Good	Good	Normal	Good	Bad	Normal
Resistant to Intense Light	Good	Good	Good	Bad	Bad	Bad
Multiple Events	Limited	Good	No	Yes	No	Yes
Advantages	Low cost (for small sizes), can detect any	Low cost (for small sizes), can detect any object	Can be deployed to a curved surface	Moderate cost, good in harsh environ-ments	Moderate cost	No overlay, superior image, can detect any object

	object					
Disadvan-tages	High costs for large areas, very sensitive to scratch and low fidelity and reduces the visibility of the screen	High costs for large areas, very sensitive to scratch, low fidelity and reduces the visibility of the screen	Delicate, expensive and sensitive to scratch	Finger activation only, reduces illumina-tion; finger must stay in position for detection, so no movement is permissible	Medium level of shock and impact resistance	Expensive, input detection placed above screen and sensitive to ambient light

Digital and film cameras share an optical system and typically use a lens with a variable diaphragm to focus light onto an image pickup device. The diaphragm and shutter lets the right amount of light into the imager – just as with film, but the image pickup device is electronic rather than chemical. Table 9: Digital camera features summarises the general features to consider when evaluating a digital camera, grouped according to their functionality [159].

2D image capture is the most common way to get an image or to record a video. Although it is sufficient for many applications (see Chapter 4.4), current image capturing and processing technologies support 3D imaging, which can be applied to different fields.

2D cameras are more common, but different kinds of 3D cameras are also available on the consumer (games) market

3D images can be captured in different ways, which can be classified according to the (number of) image devices used as well as the founding technology. 3D camera classes are introduced in Table 10: Camera technologies.

Table 9: Digital camera features

Features Group	Description and Detailed Features
Optics	Focal length (minimum, maximum); optical zoom/digital zoom; maximum aperture; autofocus (contrast detection, multi-area, centring, tracking, single, continuous, face detection, life view), autofocus assist lamp; manual focus, number of focus points; lens mounts; focal length multiplier
Sensor	Sensor size and technology; image resolution: maximum resolution and other resolutions; image ratio; colour space and filter array.
Image	Supported ISO; white balance pre-sets and custom balance; image stabilisation (optical, sensor shift); uncompressed format (raw/tiff); JPEG quality levels; file format

Photography Features	Minimum/maximum shutter speed; aperture priority/shutter priority; manual exposure mode; subject/scene mode (portrait, low-light, smart shutter etc.); built-in flash, flash range, flash modes (auto, on, off); continuous drive; exposure compensation.
Videography Features	Resolutions; formats; microphone; speaker

4.4 Computer Vision

CV deals with the interpretation of what is captured on an image (either a photo or a video).

Computer vision (CV) is a field that includes methods for acquiring, processing, analysing and understanding images and high-dimensional data from the real world in order to produce numerical or symbolic information. Duplicating the abilities of human vision by electronically perceiving and understanding an image has been a running theme in the development of this field. This understanding of the image can be seen as the disentangling of symbolic information from image data by means of models constructed with the aid of geometry, physics, statistics and learning theory.

Table 10: 3D Camera technologies

3D Image Capture Method	Description
Stereo Camera	Stereoscopy is the technique of creating or enhancing the illusion of depth in an image (a stereogram) by means of stereopsis for binocular vision. Originally, a stereogram referred to a pair of stereo images that could be viewed using a stereoscope. Most stereoscopic methods present two slightly offset images separately to the viewer's left and right eyes. These two-dimensional images are then combined in the brain to give the perception of depth. A stereo camera is composed of two cameras. Their focus is at a distance equivalent to eye focus distance, and they capture the left and right images separately.
RGB-D Camera	RGB-D cameras are sensing systems that capture RGB images along with per-pixel depth information. Such cameras can be used in the context of robotics, specifically to build dense 3D maps of unstructured indoor environments under real-world conditions. Such maps have applications in robot navigation, manipulation, schematic mapping and telepresence [160]. Microsoft Kinect or the Asus Xtion are examples of sensors that provide both depth of colour and density. There are great expectations that such systems will lead to a boost for new applications in the field of 3D perception, which is particularly relevant for robots operating in unstructured environments and under real-world conditions. Other applications of RGB sensors for 3D are mapping and localisation, path planning (SLAM), navigation, object recognition and people tracking.
ToF Camera	3D Time-of-flight (TOF) [161] technology provides 3D imaging by using a low-cost CMOS pixel array together with an active modulated light source. A 3D TOF camera illuminates the scene with a modulated light source and observes the reflected light. The phase shift between the illumination and

the reflection is measured and translated into distance. Typically, the illumination is from a solid-state laser or a LED operating in the near-infrared range (~850nm), which is invisible to the human eye.
Compact construction and ease of use, together with high accuracy and a high frame rate, make TOF cameras an attractive solution for a wide range of applications, including gesturing and 3D scanning and printing.

Machine vision [64] (also called "industrial vision" or "vision systems") has become a key technology in the area of manufacturing and quality control because of manufacturers' and customers' increasing demands for quality. Using cameras mounted over production lines and cells, machine vision utilises industrial image processing to inspect products visually, read or direct them and guide robots in real time without the intervention of an operator.

CV is already being applied in industrial field applications, but it only works correctly under controlled environmental conditions

Image processing is any form of signal processing for which the input is an image. The output of image processing may be either an image or a set of characteristics or parameters that relate to the image. Most image-processing techniques involve treating the image as a two-dimensional signal and applying standard signal-processing techniques to it [63].

Although computer vision (CV) is widely used (in industrial applications), it still has many hurdles to overcome in particular in real-time applications, including background problems, skin colour and variation in lighting conditions, to name a few. Recognition time, computational complexity and robustness are some of the constraints that the system imposes [46].

The following paragraphs briefly introduce the most relevant applications areas of computer vision.

4.4.1 Recognition

The classical problem in CV is the determination whether or not the image data contain some specific object, feature or activity. Different varieties of the recognition problem are described in the literature, including object detection (one or several pre-specified or learned objects or object classes can be recognised) [51][52] and identification (an individual instance of an object is recognised).

Recognising things in an image is the main problem CV has to deal with

In detection, the image data is scanned for a specific condition. This is usually based on relatively simple and fast computations to find smaller regions of interesting image data that can then be analysed in greater detail by more computationally demanding techniques to yield a correct interpretation.

Specialised tasks based on recognition include content-based image retrieval (finding all images with specific content in a larger set of images), pose estimation (estimating the position or orientation of a specific object relative to the camera, i.e. assisting a robot arm in retrieving objects from a conveyor belt), optical character recognition (OCR – identifying characters in images of printed or handwritten text), 2D code reading (reading 2D codes such as data matrix and QR codes), facial and shape recognition technology (SRT in people-counting-systems differentiating human beings – head and shoulder patterns – from objects).

4.4.2 Motion Analysis

Once an object has been recognised, it is possible to track it in a scene

Motion estimation comprises several tasks. An image sequence is processed to produce an estimate of the velocity either at each point in the image or in the 3D scene, or even of the camera that produces the images. Examples of such tasks include: egomotion, determining the 3D rigid motion (rotation and translation) of the camera from an image sequence produced by the camera; tracking and following the movements of a (usually) smaller set of interest points or objects (e.g., vehicles or humans) in the image sequence; optical flow, to determine, for each point in the image, how that point is moving relative to the image plane, i.e., its apparent motion. This motion is the result of both how the corresponding 3D point is moving in the scene and how the camera is moving relative to the scene.

4.4.3 Scene Reconstruction

The primary goal of 3D vision is to reconstruct a real scene for analysis

Given one or (typically) more images of a scene or a video, scene reconstruction seeks to yield a 3D model of the scene. In the simplest case the model can be a set of 3D points. More sophisticated methods produce a complete 3D surface model. The advent of 3D imaging not requiring motion or scanning and related processing algorithms has led to rapid advances in this field. Grid-based 3D sensing can be used to acquire 3D images from multiple angles. Algorithms are now available to stitch multiple 3D images together into point clouds and 3D models.

4.4.4 OpenCV

OpenCV (open source CV) [148] is a library of programming functions mainly aimed at real-time CV. The library is cross-platform (CP) and free for use under the open source BSD license [147], which means it can be used freely for research and commercial developments.

OpenCV is an OSS library that provides basic algorithms for advanced image processing

OpenCV's application areas include 2D and 3D feature toolkits, egomotion estimation, facial recognition system, gesture recognition, human–computer interaction (HCI), mobile robotics, motion understanding, object identification, segmentation and recognition, stereopsis stereo vision (depth perception from two cameras), structure from motion (SFM), motion tracking and augmented reality.

To support some of the areas above, OpenCV includes a statistical machine-learning library that contains boosting (meta-algorithm), decision-tree learning, gradient-boosting trees, expectation-maximisation algorithm, k-nearest neighbour algorithm, naive Bayes classifier, artificial neural networks, random forest and a support vector machine (SVM).

It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. OpenCV was designed for computational efficiency with a strong focus on real-time applications. Written in optimised C/C++, the library can take advantage of multi-core processing. Enabled with OpenCL, it can utilise the hardware of the underlying, heterogeneous computing platform.

4.5 Gesture Recognition, Behavioural or Gesture Analytics

Gesture recognition uses CV to provide natural ways of interaction

By far the most common interaction paradigm heralding a new age of human–machine interaction is that of the gesture [14]. Gesture recognition (GR) is a topic in computer science and language technology whose goal is to interpret human gestures by means of mathematical algorithms. Behaviour- or gesture-based analytics is the automated analysis of real-world human activity captured by video systems to track human movement and gestures, assess intentions and identify specific behaviours. Gestures map directly to “user intent” without forcing the user to learn and remember operational details of commands and options. The purpose of developing a GR system is to establish direct interaction between human and computer. Automatic gesture recognition can be used to control robots, recognise sign language [45] or access functions when driving a car [47], all of which builds a sturdier bridge between machines and humans than is possible for primitive text or graphical user interfaces (GUIs).

Hand and face gestures are of special interest.

Gestures can be static, a posture or certain pose, or a dynamic sequence of postures [48]. They can originate in any movement or state of the body but commonly emanate from the face or the hand [43]. Depending on their origin, they can be classified as [44]: hand-and-arm gestures; head-and-face gestures (nodding or shaking of head, direction of gaze, raising the eyebrows, opening the mouth to speak etc.); and body gestures (tracking movements of two people/robots; recognising human gaits for medical rehabilitation and athletic training etc.).

The meaning of a gesture is determined by the body part, its path, duration and location

The meaning of a gesture typically depends on spatial and pathic information, in other words, where it occurs and the path it takes. A similar gesture-modelling classification is proposed in [50], which classifies gesture modelling as spatial modelling that considers the characteristics of posture (the gesture's shape) in the environment of HCI applications. Temporal modelling is related to the dynamic characteristic of hand gestures (the gesture's movement). Moreover, hand modelling in a spatial domain can be implemented in 2D and 3D spaces.

Different methods were proposed to acquire the information necessary for recognition gestures system. Some methods used additional hardware devices such as data glove devices and colour markers to extract comprehensive descriptions of gesture features with ease. Other methods are based on the appearance of the hand using the skin colour to segment the hand and extract the necessary features. These methods are considered easier, more natural and less costly than the methods mentioned before. Table 11: Gesture recognition methods briefly introduce other important methods.

Table 11: Gesture recognition methods

GR Method	Description
Computer Vision–Based	This approach is most suitable for real-time applications. Many image processing techniques are used to extract, segment or detect the body part (hand) from the image. These image processing operations can be divided into two big different groups: In the **approach based on appearance,** the visual appearance of the hand image is modelled with the help of the extracted feature of the (hand) image, which is then compared with a database of features of the hand; and the **approach based on 3D model** uses a 3D model descriptor to model and analyse the hand shape. 3D models can be grouped into two categories: **volumetric model** and **skeletal model**.
Coloured Markers Methods, Marked Gloves	**Coloured markers and marked gloves methods** are closely related to CV methods. These methods use gloves to track the hand and use colours to locate the palm and fingers in order to provide and extract the geometric features necessary to form the shape of the hand. This technology is simple and cheap to use, but it can suffer from occlusion.
Sensor-Based Methods	**Sensor-based methods** use measurable values like frequency shift, time or electricity and can be classified as: **electric field,** which means they use induced potential and displacement, current caused by the proximity of a human hand, to compute the position of the hand in the field; **ultrasonic Doppler effect**; and **time-of-flight (ToF),** which relies on the time an electromagnetic (acoustic of light) wave takes to travel from source to receiver.
Instrumented Glove Methods	**Also known as wired gloves or controller-based gestures methods,** these methods use sensor devices to capture the hand's position, rotation and motion. To provide exact coordinates of the location, orientation and hand configuration of the palm and fingers, this method uses magnetic or inertial tracking devices. A physical connection between the user and a computer is required, and such devices are quite expensive.
Controller-Based Methods	These methods use controllers acting as an extension of the body so that some of the gestures' movements are captured by software. Commercial examples include: Wii Remote [149]; Myo device [150]; LG Electronics Magic Wand; Loop and Scoop, which uses Hillcrest Labs' Freespace technology. All of them use MEMS accelerometers, gyroscopes and other sensors to translate gestures into cursor movement. **Audio cubes** are another example. The sensors of these smart light-emitting cubes can be used to sense hands and fingers as well as other objects nearby and to process data.

4.6 Eye Tracking

Eye tracking traces eye movements to determine a user's focus of interest in a scene

Eye tracking refers to the process of tracking eye movements or the absolute point of gaze (POG), which refers to the point on which the user's gaze is focused in the visual scene [42][89]. An eye tracker is a device that measures eye positions and eye movement.

Eye trackers measure the rotations of the eye in one of many possible ways, and the basic methods include electrooculography (EOG), scleral search coil, videooculography (VOG) or video-based eye tracking, and video-based infrared (IR) or pupil-corneal reflection (PCR). The first two are more accurate and not influenced by the (lighting) conditions of the environment, but they are classed as intrusive; the latter two are non-invasive. All methods require a frontal view of the eye. Both methods lack accuracy because of several factors such as eyeglasses. Only the final options (IR and PCR) are not hampered by bad lighting conditions.

While eye tracking has been used to statically evaluate user interfaces, MET and PET support the evaluation of user interfaces on the shop floor.

Considering the possibilities of mobility and ubiquity for the eye trackers, they can be classified as stationary eye tracking (SET)//remote eye tracking (RET) [91], mobile eye tracking (MET) or pervasive eye tracking (PET). SET systems operate without user contact and permit free head movements within reasonable limits without the loss of tracking. All possibilities have limits, because they are based on the information provided by static sensors, usually located on a screen. Their main challenges are eye-tracking accuracy, calibration drift and the 'Midas touch' problem (distinguishing the user's intentional eye input from other eye movements).

MET/PET systems track eye movement in an unobtrusive way and allow eye-based HCI that can be used pervasively in everyday life and therefore supports 24/7 analysis [93]. Location-aware mobile eye tracking (LA-Met) [90] combines eye tracking and 2D or 3D user position tracking as a way to determine the areas of interest (AOI) of a user performing a task that requires him/her to move. This information can then be used to create mobile applications that react to the user's (assumed) cognitive processes.

Eye tracking has historically been applied to interface usability evaluation (computer applications, car consoles, etc.) as a way to interact with computers (e.g. for disabled people) or a way to determine cognitive load for people under pressure when completing a task.

Figure 11: Tobii infrared eye trackers [152]

MET/PET extends the possibilities of eye tracking to study people's behaviour by analysing their eye movement in natural environments. Firstly, it can support eye-based context inference with the use of eye-movement patterns to obtain information about activities being performed and the surrounding context. Secondly, this technology can be used to create attentive user interfaces, which means an analysis of how users' eyes are fixed on different environmental targets as attention-triggers to automatically communicate with the target in order to adapt the way in which information is provided to the users.

4.7 Audio Input/Output Technologies

An audio channel is the second way humans gather information; it is less intrusive and more effective than others

The audio channel is second only to the video channel as the most important way in which humans obtain information [190][191]. Overall it is less distracting than other (particularly, visual) channels, and acoustic warning signals are more effective and immediate than visual indicators [163]. Headphones and microphones as well as their wireless versions that are connected to smart phones or tablets via Bluethooth are well-known technologies. Chapter 5.2.3 presents hearables as an innovative view of audio-augmented interaction.

The following paragraphs briefly introduce speech recognition and text-to-speech technologies as a way to provide a non-intrusive way of interaction.

4.7.1 Speech Recognition

SR can implement short commands interaction applications even if the conditions are not very not good

Speech recognition (SR) is another way in which a computer receives user input. It is also known as **automatic speech recognition (ASR)**, **computer speech recognition** or simply **speech-to-text (STT)**.

Speech can be used as input with unrecognised speech, discrete word recognition or continuous speech recognition [7]. Moreover an SR system can be "speaker-independent speech recognition”, or it can be customised to an individual speaker, which results in very complex SR solutions. Vocalisations vary in terms of accent, pronunciation, articulation, roughness, nasality, pitch, volume and speed. Speech is distorted by a background noise, echoes and electrical characteristics.

The accuracy of speech recognition varies because of the following factors: vocabulary size and confusability; speaker dependence vs independence; isolated, discontinuous or continuous speech; task and language constraints; read vs spontaneous speech; and adverse conditions. Accuracy is one of the ways to measure the SR system’s performance. It is usually rated with a word error rate (WER), single word error rate (SWER) and command success rate (CSR). A second measure of SR performance is speed, which is measured with a real-time factor [113].

A related concept is **voice recognition** or **speaker identification** [27]. It refers to the identification of speakers rather than what they are saying. The concept can be applied to the task of translating speech in systems that have been trained on a specific person's voice or it can be used to authenticate or verify the identity of a speaker as part of a security process.

Table 12: Voice recognition applications and APIs

There are several libraries implementing Speech Recognition.

Application /API	OS	Features and Comments	Languages	License	Cloud
api.ai [194]	Android, iOS, HTML5 and Cordova	The SDKs contain voice recognition, natural language understanding and text-to-speech. api.ai offers a Web interface to build and test conversation scenarios.	15, including English, German, Italian and Spanish	Free (for non-commercial use)	
Textshark [195]	All, rest API	Cloud-/API-based speech-to-text transcription. Not real-time.		Commercial	Yes
TrulyHandsfree [196]		Embedded speech recognition for wakeup and command and control		Commercial	
TrulyNatural [197]	Android, iOS, Linux, QNX and Windows	Embedded speech recognition technologies for small vocabulary command and control, embedded large vocabulary, continuous speech recognition and multi-biometric authentication.	12	Commercial	No
iSpeech[198]	Android, iOS	Embedded and hosted solutions	18, including English, Spanish, German and Italian	OSS SDK, SaaS	Yes
Sonic Cloud Online Speech[198]	All, rest API	Supports only batch processing	18, including English, Spanish, German and Italian	Commercial	Yes
Dragon Dictation	iOs	Embedded and service. Speech-to-text and voice identification.			
MeMeMe Mobile[200]		Cloud-based speech recognition		Commercial	Yes
Vocapia[201]		Speech-to-text transcription, language identification, speech-text synchronisation. On-demand batch processing		Commercial	Yes

From a technology perspective, speech recognition has a long history with several waves [29][30] of major innovations. Most recently, the field has benefited from advances in deep learning [28] and big data. A particular field of interest is **natural language question answering (NLQA)**. This technology comprises applications that provide users with the means to ask questions in plain language. A computer or service can answer it meaningfully while maintaining the flow of interaction. It combines SR technologies with text-to-speech technologies that are described in Chapter 4.7.2.

Table 12 summarises some of the relevant SR APIs, along with their provided features. Most of the mobile devices running Android OS, Microsoft Windows Phone, iOS or Blackberry OS provide voice command capabilities. In addition to the built-in speech recognition software for each mobile phone's OS, a user may download third-party voice command applications from each OS's application store.

4.7.2 Text-to-Speech

Table 13: Text-to-speech software frameworks

Text-to-speech software frameworks make it possible for computers to have human audio interaction

Product	License	Available language/s	Programming language	Operating Systems	Cloud
Cepstral [181]	Proprietary	English, Italian, German and Spanish	C/C++	Mac OS X, Windows, i386-Linux, x86-64-Linux, Sparc-Solaris, i386-Solaris	Yes
CereProc [180]	Proprietary	11, including English, German, Austrian German, Italian and Spanish	C/Python	Linux, Windows, Mac OS X, Embedded Linux, Android, iOS	Yes
eSpeak [182]	GPLv3+	40, including English, German and Spanish	C++	Linux, Windows, Mac OS X, RISC OS	No
IVONA [179]	Proprietary	51, including English German, Italian and Spanish	C/C++	Windows, Linux, Android	Yes
Loquendo [184]	Proprietary	30, including English, Spanish, German and Italian	?	Linux, Windows	Yes
Neospeech [186]	Proprietary	30, including British English, American English and Spanish	C/C++/Java	Windows, Linux, iOS, Android	Yes
Nuance Vocalizer [185]	Proprietary	36, including English, Spanish, Finnish, German and Italian	C/C++	Linux, Windows, Android	Yes
Voxygen Expressive Speech [183]	Proprietary	French, English, Spanish, Italian and German	C/C++, Java and Python	Linux, Windows, Android	Yes

Text-to-speech converts computer outputs to audio

Speech synthesis is the artificial production of human speech, which can be implemented by either software or hardware products. A text-to-speech (TTS) system converts normal language text into speech. Synthesised speech can be created by stringing together pieces of recorded speech that are stored in a database. For specific usage domains, the storage of entire words or sentences allows for high-quality output. Alternatively, a synthesiser can incorporate a model of the vocal tract and other human voice characteristics to create a completely “synthetic” voice output. The quality of a speech synthesiser is judged by its similarity to the human voice and its ability to be easily understood. An intelligible text-to-speech programme allows people with visual impairments or reading disabilities (either permanently or "**situationally impaired**"; see Chapter 3.1) to listen to written works on a home computer.

A number of markup languages have been established for the rendition of text as speech in an XML-compliant format. The most recent is speech synthesis markup language (SSML), which became a W3C recommendation in 2004. Older speech synthesis markup languages include Java speech markup language (JSML) and SABLE. Although each of these was proposed as a standard, none of them have been widely adopted.

Table 13: Text-to-speech software frameworks summarises the most relevant text-to-speech software providers and their product features. It must be noted that, within the scope of mobile devices, some operating systems have added text-to-speech features. An example is Google Text-To-Speech, a screen reader that has been available since the release of Android 1.6. It supports several languages, including English, French, German, Spanish and Italian. Text-to-speech may be used by apps such as Google Play Books to read books out loud, by Google Translate for reading translations aloud and thus providing useful insights into the pronunciation of words, by Google Talkback and other spoken feedback–based applications, as well as by third-party apps. Users have to install voice data for each language. VoiceOver and Narrator provide similar features on iOS and Windows devices.

4.8 Context-Aware Technologies

Getting the context of the user is essential in order to implement ubiquitous computing

This chapter introduces a set of technologies that can be used to establish a user’s context of interaction (a worker, in the case of FACTS4WORKERS). In the past few years, there has been a growing interest in context-aware systems, specifically those that provide location-aware or location-based information services. For us, context-aware technologies would be any data source providing information about workers’ locations, their emotional or physiological state (qualified self). The next few paragraphs present different technologies that can be used to gather this kind of information.

4.8.1 Positioning, Location and Identification Technologies

Positioning the user is the base for determining the context of interaction.

Over the past few years, positioning (also called location-aware) technologies, have emerged. They enable the design of applications with the capability to identify a user's location and modify their settings, interfaces and functionality accordingly.

A positioning system tends to comprise several physical components: (1) one or more mobile devices usually carried around or attached to a resource; (2) a communication network that supports user-to-service interaction; (3) a service and application provider to process the positioning requests; and (4) a positioning component to provide the current location. There are often two kinds of positioning components: base stations (fixed to a location) and mobile devices.

Positioning systems require known objects to which the relative position of the object of interest is measured.

The core of any localisation method relies on the real-time measurement of one or several parameters (angles, distances, etc.) that relates a mobile devices position to a base station. Each positioning system uses different kinds of signals and various techniques to determine a resource's position, depending on the technologies used. These technologies can be categorised into four groups: infrared, radio frequency, ultrasound, and inertia signals. Based on the information measured and how the position estimation is performed, we can classify positioning techniques into four groups: (1) triangulation, (2) proximity-based, (3) fingerprinting, and (4) scene analysis techniques.

The features to be considered in order to determine the applicability of a set of technologies to a particular location problem are their accuracy, precision, scope, type of determined location (geometric or symbolic), and cost. An estimated location is considered **accurate** if it corresponds – as much as possible – to the true location of the target object. **Precision** refers to the repeatability of the measurement and indicates how sharply a location can be defined in a sequence of location determinations. **The accuracy of a localisation system** could be defined by an **uncertainty area**, i.e. the location is actually defined as an ellipse (ellipsoid) around the determined location.

GPS is the most extended outdoor positioning system.

Positioning can roughly be divided into two categories on the basis of the environment in which they work best. These two categories are outdoor positioning, and indoor positioning. In outdoor environments, GPS [53], a satellite-based positioning system, is currently the most widely used. It offers maximum coverage for positioning in these environments with relatively little effort. GPS cannot be deployed indoors, because line-of-sight (LOS) transmission between receivers and satellites is not possible in such an environment. Indoor environments are more complex than outdoor environments. The next few paragraphs introduce different indoor positioning technologies and methods.

Indoor Positioning, Object Location and Identification

Indoor positioning systems locate and track objects within buildings and closed environments. These systems use wireless concepts, optical trackers, and ultrasonic techniques.

GPS cannot be used for indoor positioning.

Several classifications of indoor positioning, location tracking and identification location are possible, based on different criteria. Figure 12: Indoor localisation system classification shows one classification that considers two main categories: **Active systems** require tracked persons to carry electronic devices that send information to a positioning system; **passive systems** use passive localisation [60] where the tracked person or object does not carry any electronic devices to infer its position. The position is estimated based on the variance of a measured signal or video process.

A second classification criterion is based on the object's or the person's location in the physical or the virtual world. A **physical location** marks a place in the real world (i.e. meeting places, houses, offices, restaurants). A **virtual location** marks a place in the virtual world, for example online. The physical location class can be broken down into three subcategories: **descriptive locations** are related to geographic objects (mountains, lakes, cities, roads, countries, etc.) and have a name or identifier; **spatial locations** are a point expressed by two- or three-dimensional coordinates in a Euclidean space; and **network locations** refer to a location based on the topology of a communications network.

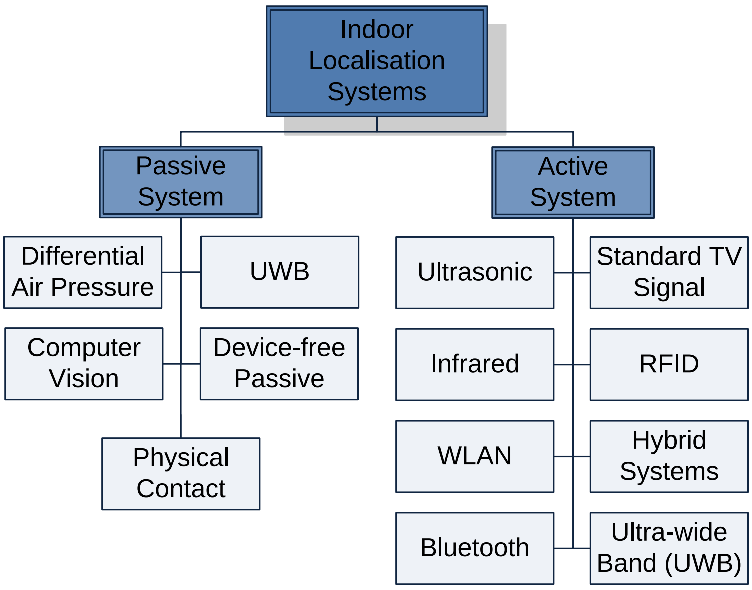


Figure 12: Indoor localisation system classification

A final classification criterion is based on the wireless communication and sensing technologies employed: **wireless communication-based localisation technologies** (Wi-Fi, Bluetooth, UWB, Zigbee, RFID, IR , ultrasound etc.) are based on wireless communication or standalone infrastructure (WLAN and WPAN); **dead reckoning localisation** technologies use motion sensors (accelerometer, digital compass, gyroscope, etc.) and odometers to determine the location of the device without the need for infrastructure; **video scene analysis** is a localisation technology based on the processing of video signals to detect specific tags in the scene (barcodes, BI-DI codes) or match the scene with pre-recorded images/videos either to determine the location of the mobile device itself or to track target objects moving in the scene. Table 14: Types of location systems summarises these methods and their main features.

Table 14: Types of location systems

Type of Location System	Description	
Dedicated	Provides a high degree of accuracy and requires expensive, dedicated equipment and is limited to small scales with high installation and maintenance costs	
	Infrared	Each tracked object is fitted with a small infrared device that emits a unique pulse signal representing its identifier. The signals are detected by at least one particular IR sensor. A location server estimates the IR device's location by aggregating data from fixed IR sensors.
	Ultrasonic	It uses an ultrasound time-of-flight trilateration technique to provide accurate physical positioning. Its main problems are NLoS (non-line-of-sight) conditions and multipath propagation in indoor environments. It achieves accuracy in the range of 1 cm–1 m.
RSSI	Estimates the distances between transmitters and receivers using radio frequency (RF) signals for triangulation or trilateration. A weakness is that walls decrease the strength of the radio signals, and the receivers perform poorly indoors. A detailed comparison of these systems can be found in [55] and [56].	
	WaveLan-Wi-Fi	It is the most popular method. It uses a radio propagation model to determine the distance to the various access points and the triangulation techniques to estimate the location of a mobile device. The advantage is that it requires only a few base stations, and it uses the general wireless networking in the buildings. The disadvantage is that the tracked object must support a WLAN network interface card (NIC). Another problem is the need to generate the RSSI radio map in advance and to update it [10]. Finally, it has to deal with different ways to measure the radio signal strength of mobile devices. It has a maximum accuracy in the range of 1–10m.
	Ultra-wideband (UWB)	It provides better positioning accuracy than Wi-Fi. UWB signals are less sensitive to multipath distortion and environment than conventional RF-based positioning systems. At bandwidths of at least 500MHz and high time resolution in the order of nanoseconds, it is possible to obtain accuracy of ranging and localisation at centimetre-level.
	Bluetooth	These systems have similar working principles as the self-localisation schemes of sensor networks. The operation principle is based on obtaining the range information to anchor devices or access points and exploring unknown device locations using trilateration. The localisation accuracy of the system is 1–5 m (<0.3 m for Bluetooth LE). It depends on the positioning technique and the characteristics (density, layout etc.) of the deployed infrastructure of devices and beacons. The emergenceof Bluetooth beacons provides new possibilities not only for indoor position but also for providing information about the objects.
	ZigBee	The localisation is usually performed using proximity and TOA methods based on distances from the surrounding ZigBee nodes calculated using RSSI achieving the accuracy of 1–10 m.

	RFID	RFID systems are designed so that the reader detects the vicinity of a tag and retrieves the data stored in that tag. Advantages include non-contact and non-line-of-sight characteristics. The absolute location of the tag is not known, but the RFID system is aware that a tag is placed within a certain distance of the reader. Besides the proximity technique, which provides the symbolic location of the tag, there are several methods for performing accurate positioning using active RFID technology. These methods employ techniques such as AOA, TDOA and RSSI, which achieve accuracy in the range of 1–5 m, or even less than 1 m (see SpotON [58] and LANDMARC [57]).
Computer Vision	Uses an alternative approach to solve the problems that dedicated and RSSI systems have: power consumption, wiring and overall permanent infrastructure cost [65]. These systems apply CV (see Chapter 4.4) to recognise fixed markers/objects features and determine where an object is located inside a building.	
	Marked-Based	They use synthetic and positioned markets to identify the object location. Based on the nature of the tracking algorithms, several types of markers can be distinguished: **ID markers or fiducial markers** are simple, geometric, 2D markers; **barcodes and quick response** codes are optical 2D representations of data items (Figure 13); **picture markers** are somewhere between ID markers and markerless tracking (Figure 14); **markerless** is the term for 2D borderless markers that do not have an explicitly rectangular boundary but have moderately textured content; **markerless 3D tracking** facilitates the detection and tracking of any real-world object using a map of 3D distinctive features (Figure 15); **CAD edge model tracking** is a tracking method that uses a 3D CAD model for an edge-based pose initialisation.
	Natural Indoor Markers	They try to avoid the inefficient and unaesthetic installation of synthetic markers all over an indoor facility. They use natural (indoor) markers, which use real-world objects, as markers. More concretely, feature descriptors of given images are saved for further recognition. Based on this feature set, they can recognise the same image from different distances, orientations and with various illumination levels, even with some occlusion, as the descriptor is unaffected by those changes. A very interesting alternative to human identification is facial recognition, which is the process of identifying humans by detecting their face and matching a scanned image of their face to an entry in a database of known samples by means of feature extraction algorithms.
(T)IMU Devices	The IMU is a single unit that houses two sensors, collects angular velocity and linear acceleration data and returns it to the main processor. It is used to acquire pitch, roll and yaw data from the device on which they are embedded. A brief description of these two sensors follows: An accelerometer is a device that measures acceleration forces that may be static, like the force of gravity, or dynamic, like vibrating or even moving the accelerometer itself. A gyroscope is a device that measures the orientation of a device based on the angular momentum of that device. It is used to acquire the angular rate of a specific vehicle.	

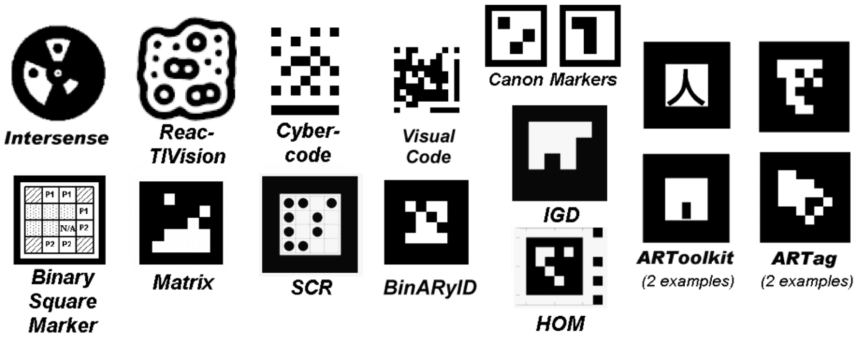


Figure 13: ID or fiducial markers

Figure 14: Examples of picture marker and markerless

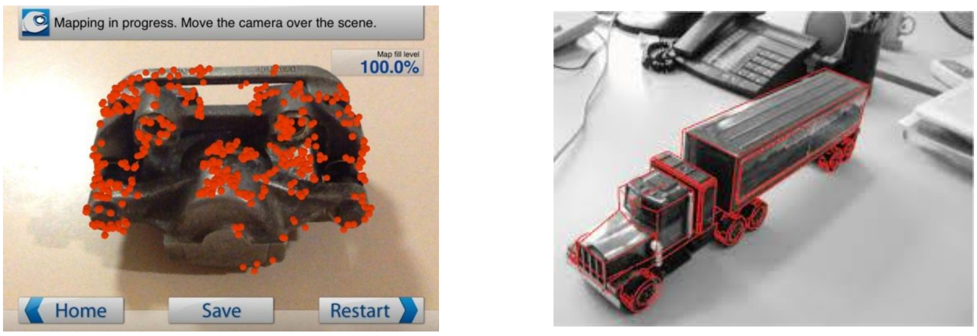


Figure 15: Markerless 3D tracking and CAD edge tracking

4.8.2 Quantified Self

Quantified self (QS) is a movement promoting the use of self-monitoring through a wide variety of sensors and devices. Applications or services based on user data about activities, biometrics, environment and experiences provide a higher level of value from wearable and mobile devices (see chapters 5.1 and 5.2), mobile apps, sensors and other services that offer self-tracking analytics, cross-sensor

QS provides information about how users take advantage of the sensors embedded in wearables

aggregation, social facilitation, observational learning and individualised coaching. Many different entities will provide these applications.

QS within the enterprise

While most of the QS-related focus has been on consumer use, it is also being discussed for use within the workplace to improve employee well-being as well as worker productivity and collaboration. The belief is that workers who are more fit and have healthier behaviours will be happier and more satisfied with their work experience, which in turn will improve productivity and collaboration. For employers, QS can also help to reduce healthcare costs. However, employee monitoring carries similar concerns which consumers have expressed regarding surveillance, security, privacy and loss of control over their personal data. In a workplace situation, the fear is that QS data will be used in performance reviews, become a requirement for continued employment or be used to justify job termination. Also, depending on the type of job and workplace setting (factory floor), management and employees may have safety concerns regarding the use of QS devices. For instance, a wearable device might become caught in machinery or create other safety hazards.

Additionally, employees might also have fashion and style concerns; for example, devices could be an unnecessary distraction if they are worn in situations with customers. However, if such employee concerns can be adequately addressed, organisations hope that QS can help to gather information about employee behaviour and influence positive improvements in business productivity. For instance, if employees have access to a personal analytics dashboard, they can be better informed and make adjustments to their behaviour and work style. Employers can gather information from personalised sensors and tracking applications collectively across the workforce, using that analytical insight to improve decision making and strategy forming related to operational efficiency, process performance and other business activities. Employers can also leverage QS as part of well-being programmes to improve employee satisfaction and retention while reducing healthcare costs. A QS project can augment employee engagement initiatives when employees do not perceive the technologies and their applications as a threat, and when they receive proper notice and there are consent agreements in place regarding how their personal data is used.

4.8.3 Emotion Detection, Affective Computing, Mood Recognition

Emotion detection uses external manifestations to determine the users' state when they are confronting a situation.

Affective computing technologies sense the emotional state of a user (via sensors, microphone, cameras and/or software logic) and respond by performing specific, predefined product/service features, such as changing a quiz or recommending a set of videos to fit the mood of the student.

Emotion modulates human communication and manifests itself through facial expression, gesture, posture, tone, vocabulary, respiration and skin physiology (temperature and clamminess). Understanding what a person is communicating entails deciphering that individual's modulation scheme, which requires knowledge of social and cultural mores and cues, as well as familiarity with the individual. Emotion recognition can only lead to optimal computing if the system is able to consider all these modes. As a result, emotion detection is going to have a very important role to play in the future development of HCI [204] – in particular, when such systems start to adapt themselves and the information they provide to the users' emotional state (see adaptive interfaces paradigm in Chapter 3.1).

Accurate emotion recognition is complex and requires the use of multi-modal algorithms in order to get accurate results.

Mood recognition technologies sense the emotional state of a user (via biometric sensors, including in fabrics like seats or clothes, cameras and interactions) and respond by performing specific, predefined actions, such as changing the lighting in a vehicle to more subtle colours to address a user's high-stress level or playing dynamic music to address driver fatigue. The technology can also be used for other applications and segments, including driver education to reduce potential accidents and improve the driving experience.

Currently there are several methods for detecting emotions by using computers. They are based on text analysis [206], voice/speech signal analysis [208] and facial/vision emotion detection [207]. All of them are at a very primitive stage of development, and they can fit different scenarios. In order to find a better solution, hybrid techniques tend to be used.

Ekman's basic emotion model is used to simplify emotion analysis

Although there are several ways to represent emotions, the most popular is the prototypical (basic) emotion categories that include happiness, sadness, fear, anger, disgust and surprise. The model, proposed by Ekman, is based on the measurement of basic emotions and supports cross-cultural nuances. The influence of basic emotion theory resulted in most studies of automatic affect recognition focusing on recognising these basic emotions. However, discrete lists of emotions fail to describe the range of emotions occurring in natural communication settings. In particular, basic emotions cover a rather small part of our daily emotional displays. A selection of categories of affect that people show in daily interpersonal interactions needs to be done in a pragmatic and context-dependent manner.

An alternative to category description is the dimensional description where an affective state is represented as a point of a set of dimensions defined by psychological concepts. One of the popular methods to describe "affective" is in terms of the dimensions of evaluation and activation. The evaluation dimension expresses how a human feels on a scale from positive to negative. The activation dimension conveys whether humans are more or less likely to take action while in the particular emotional state and ranges from active to passive. In contrast to category representation, dimensional representation enables rates to label a range of emotions. However, this projection of the high-dimensional emotional states onto a rudimentary 2D space results to some degree in the loss of information. Some emotions become indistinguishable (e.g., fear and anger), and some emotions lie outside the space (e.g., surprise). Some studies [33] use the additional dimension (e.g., dominance) to add discriminability of emotions.

4.9 Haptic Interaction

Haptic interaction sends information from the computer via physical interaction with the user

Haptic or kinesthetic communication recreates the sense of touch by applying forces, vibrations or motions to the user. This mechanical stimulation can be used to assist in the creation of virtual objects in a computer simulation, to control such virtual objects and to enhance the remote control of machines and devices (telerobotics). Haptic devices may incorporate tactile sensors that measure forces exerted by the user on the interface.

Most researchers distinguish three sensory systems related to the sense of touch in humans: cutaneous, kinesthetic and haptic. All perceptions mediated by cutaneous and/or kinesthetic sensibility are referred to as tactual perception. The sense of touch may be classified as passive and active, and the term “haptic” is often associated with active touch to communicate or recognise objects.

Applications of haptic interaction can be Teleoperators (remote-controlled robotic tools) and simulators (for operator training); video games, which are closely related to simulators (serious games) and already provided by many game controllers, joysticks and steering wheels; personal computers; mobile devices in which tactile haptic feedback is very common in the form of vibration in response to an internal event or to a screen touch; in a virtual reality environment in which adding the sense of touch to previously visual-only interfaces improves the user experience.

Because haptic interaction is performed through the skin, its development is going to be closely related to the development of smart clothing (see Chapter 5.2.4), which will allow greater progress on the haptic continuum [203] than vibrations could support today (in a mobile, on a mobile screen, with a joystick etc). Vibrations are a good way to communicate events to people under extreme conditions of light or noise.

4.10 Brain–Computer Interaction

BCI implements natural interaction but is still at an early stage of development

The idea of connecting minds with machines has long captured the human imagination. Recent advances in neuroscience and engineering are making this idea a reality and opening the door to restoring and potentially augmenting human physical and mental capabilities. Medical applications such as cochlear implants for the deaf and deep brain stimulation for Parkinson’s disease are becoming increasingly common. Brain–computer interfaces (BCIs), also known as brain–machine interfaces or BMIs, are now being explored in applications as diverse as security, lie detection, alertness monitoring, telepresence, gaming, education, art and human augmentation.

A direct brain–computer interface is a device that provides the brain with a new, non-muscular communication and control channel. A very simple model of BCI systems is presented in Figure 16. A BCI must have four components: It must record activity directly from the brain (invasively or non-invasively), provide feedback to the user, do so in real time and rely on intentional control. The latter point means that the user must choose to perform a mental task whenever he/she wants to use the BCI to accomplish a goal. Devices that only passively detect changes in brain activity that occur without any intent, such as EEG activity associated with workload, arousal or sleep, are not BCIs.

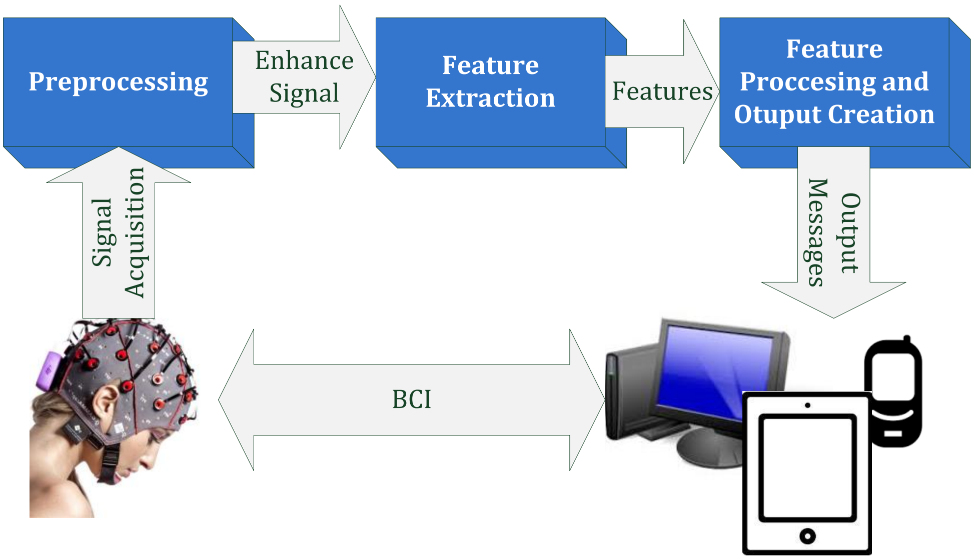


Figure 16: BCI system [226]

A conventional BCI monitors brain activity and detects certain brain patterns that are interpreted and translated to commands for communication or control tasks. BCIs may rely on different technologies to measure brain activity. A brain–computer interface is a type of user interface in which the user voluntarily generates distinct brain patterns that are interpreted by the computer as commands to control an application or device. The best results are achieved by implanting electrodes into the brain to pick up signals. Non-invasive techniques are available commercially and use a cap or headband to detect the signals through external electrodes.

A BCI can be invasive or non-invasive and can be based on electrophysiological (EEG, ECoG, intercortical recordings) or other signals, such as NIRS or fMRI. BCIs also vary in other ways, including the mental strategy used for control, interface parameters such as the mode of operation (synchronous or asynchronous), feedback type, signal processing method and application. A complete overview of BCI is offered in Annex G.

Although there are several techniques to measure performance, the most common is the information transfer rate (ITR) [25]. It depends on the number of different brain patterns (classes) used, the time the BCI needs to classify these brain patterns and the classification accuracy. ITR is measured in bits per minute. Since ITR depends on the number of brain patterns that a BCI can detect and classify reliably and quickly, the information transfer rate depends on the mental strategy employed.

Figure 17: Smart systems are supported by many other enabling technologies

5 HCI Systems

These chapters briefly introduce what we can consider to be HCI systems, namely systems that include two or more of the technologies introduced in Chapter 4 to support human interaction with the rest of the CPS and, in particular, on an industrial shop floor.

5.1 Mobile Devices

Mobile devices are hand-held computers weighing less than 900 g (i.e. tablets and smartphones).

A mobile device is a small computer, typically small enough to be handheld, with a display screen that has touch input and/or a miniature keyboard and weighs less than 2 pounds (0.91 kg) [153]. Tablet computers, smart phones and other related devices such as some smart watches and smart glasses fit this general definition. In this chapter we analyse common features to consider when a company wants to introduce tablet computers or smart phones to support workers with their daily shop-floor tasks. Smart watches and smart glasses are analysed in Chapter 5.2, because some of them cannot communicate with the cloud. Firstly, we introduce general features, and then we introduce the *ruggedisation* features to consider when trying to find a mobile device suitable for hard environments.

5.1.1 Mobile Devices Features

Figure 18 summarises mobile devices' groups of features [156][158], and the paragraphs below offer a brief introduction to them.

The first group of features concerns the ergonomic aspect of the devices. It includes their physical dimensions, **size, weight** and the grip of the device. The most common size of mobile computing devices is pocket-sized that can be handheld, but other sizes for mobile devices exist too. Mark Weiser [103] refers to devices as tab-sized or pad-/board-sized: Tabs are defined as accompanied or wearable centimetre-sized devices, such as smart phones and smart cards; pads are handheld decimetre-sized devices, such as laptops and tablet computers.

Touchscreen interface is one of the most important determinants of the device size as well as usability (see Chapter 4.2 for more details).

The next thing to consider is which **OS** the device uses. At the moment, the most relevant types of OS in the consumer area are iOS, Windows Mobile/10 and Android (see Chapter 5.3.1). Although the features they present to the user are comparable, it is important to consider how easy it is to develop and use applications on them.

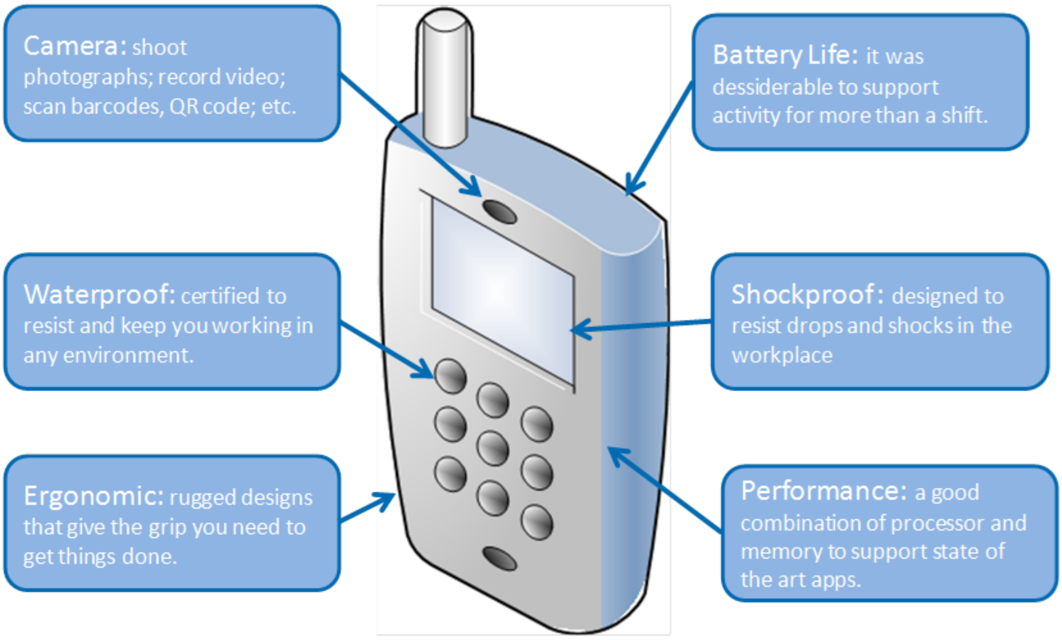


Figure 18: Summary of rugged mobile devices features [157]

Another important thing to consider is the device's supported **communication network capabilities.** At present, most handheld devices are equipped with **G4/G3**, **Wi-Fi, Bluetooth, NFC** and **hotspot tethering** (the capacity to act as a mobile hotspot that can supply Web access to nearby computers, tablets and other devices) or **infrared capabilities**. For mobile phone network–connected devices, it would also be important to consider where it is possible to use dual SIM allowing the device to be used both in for business issues as well as personal ones.

With regard to both communication and imaging capabilities, for some business applications, it would be interesting to include RFID, smart card or (laser) barcode reader capabilities. Over the past few years, because of the popularity of the different graphic codes used in logistics (see Chapter 4.8.1) to identify objects and places, several specially designed mobile devices (usually PDAs) have embedded a laser reader because of CV-based systems' lack of accuracy. At the moment, another important characteristic to consider is the camera's capabilities. Today most smart phones and tablets embed a front and a rear camera. As camera technology gets cheaper and cheaper, image capturing features (see Chapter 4.3) tend to be equalised, although some models still keep them better for the rear camera. Halfway between the hardware and software capabilities of image and video capturing, some important issues can be LED flash, 3D capture, autofocus and touch focus, dual camera record, optical image stabilisation and gesture shot, among others.

As important as their communication and interaction capabilities are, is the performance. These capabilities are supported by both the processor and the internal memory. Today, processors tend to embed more than one core (the

Samsung Galaxy S6 supports eight), which allows a better parallel execution of multithread processes and support of multi-tasking. An additional issue when aiming to present 3D graphics to the user is the graphics capabilities of the processor, for which GPU support is desirable.

Beyond the performance of the processors, the performance of the memory is another important point to consider. It can be measured by speed and based on its capacity. At present, up to 64GB of internal memory can be embedded in consumer mobile devices. It has a higher speed than external memories but, depending on the application, an expansion slot to hold microSD memories is also desirable.

Other features included in most mobile devices today include IMU support, GPS, USB (as a way to charge the battery or to exchange information) and USB OTH (USB on the go) is a standard that enables devices to talk to one another).

Table 15: A few reference sites from which to choose mobile devices

Mobile Device Type	Comparison Site
Smartphone Place	http://www.91mobiles.com/compare/Samsung/Galaxy+Grand.html http://www.geek.com/wp-content/uploads/2014/12/2014-12-17-11_19_27-Gnod-Smartphone-Comparison-Chart.png http://smartphones.specout.com/ http://www.productchart.com/smartphones/
Rugged Smartphone Place	http://www.scandit.com/2015/03/20/rugged-smartphones-in-the-enterprise-overview-and-2015-buyers-guide/ http://smartphones.specout.com/d/p/Rugged-Smartphone
Tablet Place	http://www.phonearena.com/phones/Class/Tablet http://www.tabletpccomparison.net/ http://www.productchart.com/tablets/
Rugged Tablet Place	http://www.ruggedpcreview.com/2_comparison_tablets.html http://www.groupmobile.com/compare.asp

The batteries are the basis for the functionality of all mobile devices. The capacity (mAH), the working time (talk time, standby time), the material used to carry electricity (Li-ion), the probability to remove it and the possibility to charge it wirelessly are all features to consider. Table 15: A few reference sites from which to choose mobile devices provides a set of references to sites where the latest mobiles features are compared with each other.

5.1.2 Rugged Mobile Devices

Rugged IT is a marketing term for hardware that is designed to operate in extremely harsh environments and conditions [154]. There are three generally accepted levels of *ruggedisation*: **semi-rugged, fully rugged and ultra-rugged**. The levels describe a product's ability to survive drops, vibration, dust, immersion and extreme temperatures.

Semi-rugged devices, which are increasingly being called business-rugged, are usually enhanced versions of commercial off-the-shelf (COTS) hardware. The components are the same, but they are better-protected, for example, a smart phone with a **thicker case**. Fully rugged devices are designed from the inside out to work in extreme temperatures, to be impervious to being dropped, to resist shocks and vibrations and to be dustproof and waterproof. A fully rugged laptop may have a solid state hard drive, which has no moving parts and doesn't need a fan. Ultra-rugged devices, which are usually designed to meet precise specifications for military use, are made to cope with the harshest environmental conditions. An ultra-rugged laptop can be left out in a sandstorm, frozen in a blizzard or sent on a vibrating rocket into space without any detrimental effects.

***Ruggedisation* levels are not standardised, but most providers refer to the Ingress protection code or the MIL STD 810.**

Table 16: IP suggested levels of protection for several industrial scenarios [157]

Industry/User Scenario	MIL-STD-810G	IP54	IP68
Logistics – pickup and delivery	✓		✓
Logistics – order management	✓		✓
Manufacturing – asset management	✓		✓
Various industries – field service	✓		✓
Retail – point of sale	✓	✓	
Retail – clienteling	✓	✓	
Retail – procurement	✓	✓	
Retail – inventory management	✓	✓	
Health care – patient beside care	✓		✓
Health care- pharmacy applications	✓	✓	
Government – document tracking	✓	✓	
Government – military applications	✓		✓

The levels of *ruggedisation* are not standardised, which means that vendors have the freedom to use the labels as they see fit. Most vendors incorporate other values into their evaluations to provide potential customers with some assurance that their products deserve the label they have given them. The two most common values cited are from the Ingress protection (IP) code [155], a system for classifying the degrees of sealing protection provided by the enclosures of electrical equipment, and MIL-STD-810 [157], a series of testing guidelines set by the U.S. Department of Defense for military and commercial equipment.

Table 16: IP suggested levels of protection for several industrial scenarios [157] confronts some usual industrial scenarios to the MILSTD810G standard and the IP54 and IP68 levels of protection.

IP codes use two digits [240]. The first number represents the protection against dust. It takes values in the range 0 (No protection) to 6 (Totally protected). In the table it is used the 5th level meaning that the devices should be protected against dust with limited access (very thin dust particles). The second number represents the Protection level against water. It takes value between 0 and 8. The 4th level means the device is protected against splashing water, while the 6th level requires the device should have to be protected against a nozzle under pressure.

Finally a MIL-STD-810 compliant device, because of the test severity and range may be used for any industrial scenario.

5.2 Wearable User Interfaces

Wearables provide unobtrusive, contextualised HCI

Wearable user interfaces describe the interaction between humans and computers through electronics designed to be worn on the body. They may sense the human body or the environment around the wearer and transmit the information to a smart phone or to the cloud. Ideally, wearable user interfaces are unobtrusive, always switched on and wirelessly connected, and they provide timely information in context. Examples of wearable electronics are smart watches, smart glasses, smart clothing, fitness monitor wristbands, sensors on the skin and audio headsets.

Smart fabric refers to a range of technologies that transform material used in clothing, upholstery and other textiles into devices that can be deployed as sensors, switches, connectors, batteries and displays. The components and electronics may be embedded into the fabric or even, in some cases, into the fibres.

5.2.1 Smart Watches

A smart watch is a computer attached to the user's arm

A smartwatch is a computer attached to the user's arm. At the moment there are three different tendencies in smartwatch implementation [83]: Some smart watches are independent computers, such as the Pebble smart watch; there are also manufacturers that provide smart watches, such as complementary interfaces for mobile devices (smart phones or tablets), for example Sony or Samsung; finally, there are manufacturers that adopt a hybrid approach by connecting the smart watch to the smart phone to obtain applications and data while being able to operate independently, and manufacturers adopting Android Wear, the Apple Watch and the Microsoft Band follow this philosophy.

While there are big expectations for their use in the future, smart watches currently suffer from some major constraints that reduce their introduction, in particular in the industrial world. Firstly, they have a small screen size, which means restricted I/O. Consequently, it is difficult to fit a keyboard onto it or display multimedia. Several solutions have been proposed to solve this problem. Micro-projectors can be a solution, but they are not wearable. A larger (curved) screen could be another solution to the problem. Additionally, ways need to be found to ensure they are lighter, waterproof and anti-scratch. Moreover, a new user interface design and new interaction techniques such as 3D ultrasonic gestures recognition or voice input (i.e., Android Wear) should be developed.

A second constraint (more precisely, a set of constraints) is related to smart watches' small hardware, which means less computing power, smaller battery capacity and less precise sensors. Although hardware components get smaller thanks to advances in sensor technologies and electronics, the smaller footprint means smart watches have fewer ubiquitous computing features. Most of the smart phone–dependent smart watches overcome this hurdle by offloading power-consuming sensing and computing operations to the phone and using low-power Bluetooth to communicate. This enables smart watches to communicate with the smart phone and to rely on its superior computing capabilities: They simply use the watch as a convenient user interface.

Known position and contact with the skin are two of the advantages of smart watches

However, smart watches also have numerous advantages. Firstly, they are body-mounted and have a standard, known location. Secondly, and probably more importantly, the continuous connection to the skin allows them to recognise their owner's physical activities and location, which can be exploited by mHealth applications. Because of the location, it is also easy for them to record the heart rate, heart rate variability, temperature, blood oxygen and galvanic skin response (GSR). GSR can be used to identify physiological arousal.

Evaluation Criteria

Comparison criteria of smart watches

The previous paragraphs briefly introduced smart watches' types, constraints and advantages. Now we provide some criteria to use in order to determine whether or not they should be adopted in a project. First of all, a few implementation features need to be taken into consideration. These general features include anti-loss alert, time display, call vibration, caller ID, answer call and micro USB input port. Additional criteria, introduced in [82], are expanded on below.

- **Platform (OS):** The first criteria to consider can be divided into two: the compatibility of the device with the supporting device (phone or tablet) and the platform/OS that it uses.

Some operating systems to consider for second screen and hybrid devices if using a complementary phone include Android, iPhone, Windows Phone and Blackberry.

OS platforms currently running on smart watches include [84][85][86]: Brand by Microsoft, Android Wear, Apple Watch (requires pairing with iOS 8.2+, based on a subset of Apple iOS), Connect IQ, Firefox OS for Wearables by Mozilla, LinkIt OS by MediaTek, Linux Derivative, Pebble, Tizen for Wearables by Samsung and WebOS by LG. On the basis of current market penetration and the number of available apps, it seems Android Wear and Apple Watch OS will be the most popular systems [86].

The most common kinds of displays on smart watches are E-Paper, Retina, LCD, OLED and AMOLED.

- **Screen:** Smart watches are mostly used for reading notifications rather than watching videos, browsing the Web or viewing photos. Even though the screen size is only about 1.5 inches, the display needs a fairly high resolution in pixels per inch in order to provide sharp and easy-to-read fine text and graphics. It also needs to produce fairly bright images, because watches are often viewed in high ambient light. A larger colour gamut is also required in order to counteract colour washout from ambient light, and vibrant saturated colours are quite helpful when reading screens with text and graphics information. A major challenge is accomplishing all of this and have more than one day of battery running time.
- **Hardware**: As the smartwatch world moves forward, so do the technologies that power them. Whereas phones need big processors and lots of RAM to work well, smart watches just need to work. Smart watches do not need a whole lot under the hood to work well, with the exception of phone watches that run a full OS.
- **Software:** For watches working as secondary screens, the software is not of any importance, as the devices simply relay notifications from your phone. However, the software has to be taken into consideration with independent computers and hybrid smart watches.
- **Battery:** Quite possibly the most important factor when choosing a smart watch, battery life varies almost as much as the watches themselves. While it is accepted that phones and tablets have to be recharged on a daily basis, this is not the case for smart watches. A single charge should last at least two or three days. Some simple software updates should still improve the battery life.

Smartwatches Comparison

Once we have defined the criteria, we perform a review of smart watches currently on the market, based on the revision presented by [88]. Table 17: Smartwatches comparison in November 2015 provides a summary.

Table 17: Smartwatches comparison in November 2015

	 Apple Watch	 Alcatel One Touch Watch	 Motorola Moto G 360	 LG G Watch R	 Sony Smart watch3	 Asus Zen Watch	 Huawei Watch	 Samsung Gear S2
Smartphone Compatibility	iPhone 5+	iOS7+, Android 4.3+	Android 4.3+	Android 4.3+	Android 4.3+	Android 4.3+	Android 4.3+	Android 4.3+
Operating System	Watch Os	Custon Android	Android Wear	Android Wear	Android Wear	Android Wear	Android Wear	Tizen
Display Type	Retina	LCD	LCD	OLED	LCD Trans-flective	AMOLED	AMOLE D	SUPER AMOLED
Screen Size	1.5"-1.65"	1.22"	1.56"	1.3"	1.6"	1.64"	1.4"	2"
Screen Resolution	340x272 390x312	204x204	320x290	320x320	320x320	320x320	400x400	480x360
Battery (mah)	205	210	320	410	420	370	300	300
Sensors	Heart rate, pulse oximeter, accelerometer, gyroscope barometer ambient light, force touch	Heart rate, accelerometer, gyroscope, altimeter	Heart rate, pedometer, nine-axis accelerometer, ambient light	Heart rate monitor, barometer, accelerometer, gyroscope	ambient light, accelerometer, compass, gyroscope, GPS	Heart rate, accelerometer, gyroscope, barometer	Heart rate, barometer, gyroscope, barometer	multi-touch, accelerometer, gyroscope compass, heart rate, ambient light, UV sensor, barometer
CPU	Apple 5.1	STM-429	Cortex A8 1GHz	Qualcom Snap-dragon Quad-Core 1.2GHz	ARM A7 Quad 1.7GHz	Qualcom Snap-dragon Quad-Core 1.2GHz	Qualcom Snap-dragon Quad-Core 1.2GHz	Qualcom Snap-dragon Dual-Core 1GHz
Storage	8GB	4GB	4GB	4GB	4GB	4GB	4GB	4GB
Bluetooth	✓	✓	✓	✓	✓	✓	✓	✓
NFC	✓	✓			✓			✓
Wi-Fi	✓		✓	✓	✓			✓
Microphone	✓	✓	✓	✓	✓	✓	✓	✓
Speaker	✓							✓
Waterproof	Water-resistant	30' at 1.5 m	30' at 1.5 m	30' at 1.5 m	water-proof	water-resistant	30' at 1.5 m	30' at 1.5 m

5.2.2 Smart Glasses

Smart glasses are HMDs that provide unobtrusive visual access to information

Smart glasses are a kind of see-through head-mounted display. **See-through HMDs** are mostly employed to allow the user to see the real world with virtual objects superimposed onto it by means of optical or video technologies. They may be fundamentally divided into two categories: optical see-through (OST) and video see-through (VST) HMDs.

Optical see-through (OST) allows the user to see the real world naturally with his/her eyes while overlaying graphics onto the user's sight by using a holographic optical element, half-silvered mirror or similar technology. Its main advantage is that it offers a view of the real world with overlaid digital information.

Video see-through (VST) displays are those in which the user has a video view of the real world with graphics laid over it. The advantages of VST HMDs include consistency between real and synthetic views and the availability of a variety of image processing techniques, like correction of intensity and tint, and blending ratio control. Thus, because of various image processing techniques, VST displays can handle occlusion problems more easily than OST displays, but correct occlusion effects between virtual and real objects is the main issue to be resolved.

Smart glasses features for comparison

Smart glasses usually provide information to the user using video-see-through. Many of them are currently available on the market. Table 18 presents an overview of the most relevant ones that fit the following criteria:

- The HMD has to augment the user's experience in some way. Units that are only capable of shooting photos or video were eliminated, as were units designed exclusively for virtual reality (VR).
- There had to be at least a functional prototype in existence with an intention to work towards production.
- Devices that were not designed to be head-mounted displays (use cameras for input and VDT for output) were also eliminated.
- The list is composed of a subset of these HMDs whose intended use is for the mobile consumer, and it evaluates their viability and competitiveness.

For the selection of concrete devices, criteria can include [78][79]:

- **Cost** – always an issue and trade-off.
- **Size and weight** – weight distribution and weight on the person's nose.
- **Resolution** – more is better, but it has a proportional cost for both the display and optics.

- **Field of view (FOV)** – a wider FOV is more immersive and supports more information, but to support a wide FOV with good angular resolution throughout and support high acuity would require an extremely high resolution display with extremely good optics, which would be extremely expensive even it possible.
- **Exit pupil size** – How big is the sweet spot for viewing the image in the optics? If you have ever used an HMD or binoculars, you will have noticed how you have to get them centred right or you will only see part of the image with dark rings on the outside. As the FOV and eye relief increase, it becomes more and more difficult and expensive to support a reasonable exit pupil.
- **Vision blocking (particularly peripheral vision)** – this can be a serious safety consideration for something you think you would wear when walking and/or driving.
- **Eye relieve and use with glasses –** how far away from the eye is the last optical element? This is made very complicated by the fact that some people wear glasses, and faces have very different shapes.
- **Vision correction** – many people do not have perfect vision, and generally the HMD optics have to be in the same place as a person's glasses. Some designs have included dioptre/focus adjustment, but many people also have astigmatism.
- **Adjustment/fit** – this can open a can of worms, as the more adjustable the device is the better it can be made to fit, but subsequently it becomes more and more difficult to get it to fit properly.
- **Battery life (and weight)** – the battery has to be moved either to the back of the head or via a cable to some place other than the head.
- **Connection/cabling** – wireless means severe compromises in terms of power, weight, support on the head and processing power (heat, battery power and size).
- **How it is mounted –** head bands, over the head straps, face goggles etc.
- **Appearance** – the more you try to do on the head, the bigger, bulkier and uglier it is going to be.
- **Storage/fragility** – HMDs generally do not fold up into a very small form factor/footprint, and they are usually too fragile to put in your pocket.
- **Kind of device : see-through; monocular** (one-eye); **binoculars** (both-eyes); **centred vertically –** the display will tend to dominate and even, in the case of non-see-through, totally block the user's vision of the real world; monocular displays are located **above or below** the eye so that they do not impair forward vision when the user looks straight forward. This is not optimal for extensive use and can cause eye strain. The positions above and below are better for "data snacking" rather than long-term use.

- **SDK** – do they provide a SDK?
- **Operating system –** which OS are they running?
- For which **market** have they been designed? Industrial, military, medical, consumer.
- Additional characteristics to be considered: camera quality and sensors (IMU).

Table 18: Comparison of smart glasses features **shows** the more relevant smart glasses to be used on the industrial shop floor. This comparison is based on the FACTS4WORKERS, and a complete comparison is available at [235].

Smart glasses open new possibilities for industrial applications. This is why we took a closer look at the smart glasses that are currently available and created a comprehensive market overview, available on our website:
http://facts4workers.eu/smart-glasses-comparison/

Table 18: Comparison of smart glasses features

Modell		Daqri Smart Helmet	EPSON Moverio BT-200	GlassUp	Kopin Golden-i Gen 3.8	Laster Pro Mobile Display	Lumus DK-40	Meta Pro	Optinvent ORA-1	Recon Jet	Sony SmartEye glass	Vuzix M100	Vuzix M2000AR
Software	OS	Android Based	Android 4.0.4	Onboard OS, Android, iOS via Smartphone, soon Windows	Android		Android	open, OS running on wearable computer	Android 4.2.2	Android	Android 4.4+	Android 4.0.4	
	SDK	yes	yes	yes	yes		yes	yes	yes	yes	yes	yes	
Display	Display Type	OST	OST	OST	VST (Micro-Display)	OST		OST	OST	VST (Micro-Display)	OST, monochrome green	VST (Micro-Display)	OST
	Monokular/Binokular	binocular	binocular	monocluar (right, in the center field of vision)	monocular (left or right)	monocular (right), image in the central field of view	monocular (right), image in the central field of view	binocular	monocular (bottom or centrally adjustable)	monocular (right)	binocular	monocular (right or left)	
	Resolution		960x540	320x240	960x540	800x600	640x480	1280×720	33 px / degree	428x240	419x138	432x240	1280x720
	FOV		23°			40° x 30°	25°	40°	24°			15°	30°
	Virtual Image Size		40" from 8 ft away		15"	30 cm to 3 m				30" from 7 ft away		4" from 1 ft away	13"-24"
Interaction	Specials	Through and smart watch like interface	Touchpad (separate)	Touchpad (frame)	Speech, Gestures		planned for the future with eyeSight gesture control	Gesture	Touchpad (frame)	optical touch sensor	Touchpad (separate)	Speech, Gesture, 4 Buttons	4 Buttons
Hardware	Battery (mAh)	more than a shift	2720 mAh (6 h)	24 h	8 h, changeable		Up to 4 h	4-6 h	3-8h	4-6 h	without camera 150 min, with camera 80 min	2 h	2-3 h
	Camera	2 ahead, 2 behind, high resolution depth sensor	VGA	no	14 MP (IR camera optional)	yes		twin RGB cameras	Photo: 5 MP Video: 1080p	720p	yes	Photo: 5 MP Video: 1080p	5 MP

	Speaker/Microphone		yes		yes				yes	yes	yes	yes	yes	optional
	Microphone		yes		yes					yes	yes	Yes, noise suppression	yes	optional
Interfaces	USB		yes	yes	yes			yes	yes	yes	yes, only charging	yes	yes
	WiFi		yes		yes	yes (optional)		yes	yes	yes	yes	yes	
	Bluetooth		yes	yes	yes			yes	yes	yes	yes	yes	
	Other		yes		LTE					ANT+			
Sensors	GPS		yes	via Smartphone	yes				yes	yes		yes	
	Gyroscope, Accelerometer, Compass		G.A.C	G.A.C	G.A.C			G.A.C	G.A.C	G.A.C	G.C	G.A.C	G.A.C
	Other		yes		thermal image through IR camera				brightness sensor	yes	brightness sensor	proximity, brightness, gesture	
Product	Weight		88 g	70 g	185 g (depending on equipment)	130 g		180 g	70 g	Temperature	77 g without controller (controller 45 g)		
	Price		699 USD	299-499 USD		5000 EUR		3650 USD	949 USD	60 g	670 EUR	1000 USD	5999 USD
	Availability		available	pre-order				pre-order	pre-order	699 USD	available	available	
	Market	Industrial	Consumer		Industrial	Industrial	Consumer	Consumer	Consumer	available		Industrial	Industrial
	Specials	depth sensor, 360° navigation cameras, especially for industrial use	Separate Touchpad	"cheaper" Google Glass version, smaller resolution, fewer sensors, mainly an output device	especially for police, fire brigade, military	color depth: 24-bit	connected with smartphone via wireless	MetaPro have their own app store, MetaApp Store where 500+ apps are under development. Needs a Wearable Computer.	AR mode (central front of the eye) and dashboard mode (below the field of view)	Consumer	Connected with BT or WiFi via Android Smartphone for working with apps		Ruggedized display and camera housing

5.2.3 Hearables

Hearables provides better access to health data than smart watches

Hearables are wireless wearable computer-like earpieces that enable voice input and return results as audio into the user's ear. Hearables are a new wave of hybrid devices that merge the health tracking capabilities of a smart watch with the high-quality audio we have come to expect from premium earbuds [165]. They provide better readings of heart rate and body temperature than other devices, because they are more accurate coming from an in-ear device than a device attached to the body [163], like a bracelet or a smart watch [166].

Controlled by touch, movement, thought or voice (or any combination of these), the miniaturised in-ear computers are designed primarily for the purpose of mobile communication, real-time information services, activity tracking and various monitoring applications focusing on the wearer's health conditions and body performance, mostly in combination with a wireless media player [164].

It is less distracting to provide audio information than via AR tools, wearables or mobile phones

One important benefit of placing the entire interaction unit in the ear and addressing users purely through acoustic signals is that there is less distraction in comparison with vision-based augmented reality tools or wearables with tactile signal, measuring and interaction mechanisms. At the same time, as experiments have shown, acoustic warning signals are more effective and immediate than visual indicators. Furthermore biometric data such as temperature, heart rate or oxygen saturation can be measured and monitored with significantly higher reliability and better response times through in-ear monitoring than contact devices placed on the wrist or the torso.

For the hearing aid industry, hearables are likely to be a disruptive technology, because they could perform hearing aid tasks better, with additional functions, at a lower cost and with less user-perceived social stigma. But they also offer opportunities for those with normal hearing as well. For example, this technology could help the increasing number of elderly people to avoid using conventional input/output devices such as a keyboard, mouse or touch screen.

The two big players to keep an ear out for (!) are Valencell and Bragi. Valencell develops the sensors that enable heart rate tracking in the current first-to-market hearables, like the Jabra Sport Pulse and SMS Audio Biosport. Bragi is the company behind the widely touted Dash, the most sensor-laden hearable of them all. Other companies working in the development of hearables are Elbee, Waverly Labs [170], Motorola ("Moto Hint" already offers a hands-free, voice-controlled earbud for Bluetooth-enabled smart phones) and Apple. Table 19: Hearable Features Comparison compares the features of some of the most relevant hearables with each other.

Table 19: Hearable Features Comparison

Comparison of hearable features

Feature	Elbee[168] 	The Dash[169] 	Moto Hint[171] 
Producer	Elbee	Bragi	Motorola
Status	In beta	In production	In production
SDK	Yes	Not available	Not available
App Customisation	Yes	Yes (under development); iOS Android, Windows Mobile	Yes (iOS, Android)
IMU	Nine-axis accelerometer, two movement processors	Three-axis accelerometer, three-axis gyroscope, three-axis magnetometer	Not available
Interaction	Capacity sensors, voice, proximity sensor on/off	Capacity sensors, voice, proximity sensor on/off	Proximity sensor on/off
Battery	100mAh Li-Ion, 3h duration, storing and charging chase +12h	100mAh Lithium polymer battery, 3h (play + track), standby 250h, duration, storing and charging chase with five charges, USB	Talk time: Moto Hint: up to 3.3 hours; Moto Hint and charging case: up to 17 hours. Standby: Moto Hint and charging case (with Moto Voice disabled): up to 200 hours
Weight	10 g	Not available	Not available
Connectivity	Bluetooth 2.1 + 4.0	Bluetooth 4.0, Bluetooth LE	Bluetooth® enabled A2DP 1.3, HFP 1.6, AVRCP 1.4, HSP 1.2; PBAP 1.1; Generic Audio Visual Distribution Profile (GAVDP) 1.2; Bluetooth® Technology 3.0 + EDR
Communication	Two devices, active noise reduction, voice triggering, speech recognition, text-to-speech	Two devices, active noise reduction, audio transparency (ambient sound)	One device, noise reduction and echo cancellation; volume and mute controlled through phone; advanced multipoint - simple pairing of secondary devices; pass through audio
Additional Capabilities	Home automation	4GB internal storage, heart rate, steps, duration, 1 m waterproof	

5.2.4 Smart Clothing

Smart clothing incorporates electronic both for gathering information and providing information from/to the wearer

Smart clothing, also known as e-textiles, are fabrics that embed digital components (including small computers), and whose electronics add value to the wearer [177]. What makes smart fabrics revolutionary is that they have the ability to do many things that traditional fabrics cannot, including communication, transformation, conducting energy and even growing. Smart clothing is distinct from wearables because the emphasis is on the seamless integration of textiles with electronic elements like microcontrollers, sensors and actuators. Furthermore, e-textiles need not be wearable. For instance, e-textiles are also found in interior design.

Smart textiles can be broken down into two different categories: aesthetic and performance-enhancing. Aesthetic examples include everything from fabrics that light up to fabrics that can change colour. Some of these fabrics gather energy from the environment by harnessing vibrations, sound or heat, and reacting to this input. Then there are performance-enhancing smart textiles, which will have a big impact on the athletics, extreme sports and military industries. There are fabrics that help regulate body temperature, reduce wind resistance and control muscles. Other fabrics have been developed for protective clothing to guard against extreme environmental hazards like radiation and the effects of space travel. The health and beauty industry is also taking note of these innovations, which range from drug-releasing medical textiles to fabrics with moisturising, perfuming and anti-aging properties.

Some examples of the many uses of smart clothing include: health monitoring of vital signs of the wearer such as heart rate, respiration rate, temperature, activity, and posture; sports training data acquisition; monitoring personnel handling hazardous materials; tracking the position and status of soldiers in action; monitoring pilot or truck driver fatigue; innovative fashion (wearable tech); regaining sensory perception that was lost by accident or at birth.

5.2.5 Nearables

Nearables are very small computing objects that are at the core of the IoT

Nearables is a term now commonly used to describe the idea of smart objects: everyday items with small, wireless computing devices attached to them. This is very much a companion piece to wearables. Nearable technology can communicate with mobile devices within range and offer a significant and diverse range of useful information [172].

In its first meaning, nearables are not devices themselves. Any object (or a live being, like a human or animal) can become a nearable after a wireless, electronic sensor is attached to it and starts broadcasting data to nearby mobile devices. Owing to the continued miniaturisation of sensor technology, a single transmitter could be equipped with a whole set of these, for example an accelerometer, a thermometer, an ambient light sensor, a humidity sensor or a magnetometer. In the second meaning, nearable devices can be part of an infinite array of smart, interconnected objects programmed to improve an individual's surroundings in every way, usually in a smart home environment.

The first examples of nearables were objects tagged with Bluetooth Smart beacons supporting accelerometer and temperature sensor and broadcasting their signal in a range of approximately 50 meters. They can communicate with mobile applications installed on devices with Bluetooth 4.0, compatible with Bluetooth Smart protocol on the software side.

iBeacon and Eddystone are two examples of nearable technology

There are already many examples of nearable technology, with iBeacon [174] an obvious and prominent one. iBeacon is a communication protocol developed by Apple on top of Bluetooth Smart technology that uses Leveraging Bluetooth Low Energy (BLE)[173]. Each object is attached to an iBeacon compatible device that is identified by an iBeacon (hierarchical) identifier. The objects broadcast tiny packets of data that contain their iBeacon ID and information about signal strength, so that the phone can understand which beacon it hears and how far it is. iBeacon allows for two basic interactions between apps and individual beacons or groups of beacons called “regions”:
Region monitoring: Actions are triggered upon entering/exiting the region’s range, a and it works in the foreground, background, and even when the app is switched off.
Ranging: Actions are triggered based on proximity to a beacon. It works only in the foreground.

Eddystone [175] is an open Bluetooth 4.0 protocol from Google. Although it provides similar features to iBeacon, there are some differences between the two [176]: iBeacon is officially supported only by iOS devices, while Eddystone has official support for both iOS and Android; Eddystone is an open protocol; Eddystone is designed to support multiple data packet types (Eddystone-UID; Eddystone-URL; Eddystone-TLM); iBeacon provides two API methods for apps to detect iBeacons devices, namely ranging and monitoring.

As both protocols are based on BLE protocols, they require that an application understand the broadcasted packet data.

5.3 Cross-Platform (CP) Software Environments

CP software is able to run on more than one platform

In computing, **cross-platform (CP)**, or **multi-platform**, is an attribute attached to computer software or computing methods and concepts that are implemented and inter-operate on multiple computer platforms [80]. The software and methods are also said to be **platform-independent**. CP software may be divided into two types:

One requires individual building or **compilation** for each platform that it supports, while the other can be run directly on any platform without special preparation, e.g., software written in an interpreted language or **pre-compiled portable bytecode** (like Java) for which the **interpreters or run-time packages** are common or standard components of all platforms.

The term **platform** can refer to: **the type of processor and/or other hardware** on which a given OS or application runs; the type of **OS** on a computer; or the **combination of the type of hardware and the type of OS running** on it as well as the software components that make it possible to develop additional services and extensions (to the provided by the OS). In the case of smart mobile devices, the mobile platform consists of the OS, the necessary hardware components, and the software development kits (SDK) provide the necessary tools and resources for the development, installation and test of the applications.

Platform can refer to the type of hardware, the type of OS or a combination of both

For **a piece of software to be considered CP**, it must be able to function on more than one computer architecture or OS. Developing such a programme can be a time-consuming task, because different OSs have different application programming interfaces (API). That is why using a CP SDK can reduce development efforts. The following paragraphs evaluate the most relevant mobile and wearable OSs as well as the available CP development frameworks to develop applications for them.

5.3.1 Most Relevant Wearable/Smart Operating Systems

A mobile OS (or mobile OS) is an OS for smart phones, tablets, PDAs or other mobile devices [140]. Mobile OSs combine the features of a personal computer OS with other features useful for mobile or handheld use; usually including (and most of the following are considered essential in modern mobile systems): a touch screen, cellular, Bluetooth, Wi-Fi, GPS mobile navigation, camera, video camera, speech recognition, voice recorder, music player, near field communication and infrared blaster.

Mobile OSs abstract the access to hardware embedded in the smart device

Current versions of these software platforms are Android 5.1.1, iOS 9.x, Windows Phone 8.1, BlackBerry 10.3, Firefox OS 2.5.0, Sailfish OS Update 16, Tizen 2.3.x and Ubuntu Touch OS. Table 20: Worldwide smartphone sales to end users by OS in 2Q15, shows the result of a forecast analysed by Gartner [141].

According to this table and [138], we see that Android, iOS and Windows Phone are the three major players on the OS market. A complete comparison of their features and capabilities can be found at [139]. Their most relevant features for the FACTS4WORKERS project scope are: GPU accelerated GUI (to support AR applications, for example), per-application feature access; common APIs for smart phones, tablets etc.; official multi-platform SDK platforms; basic features; browsers;

The most popular mobile OSs are Android, iOS and Windows

language (non-English language support) and inputs (gesture text input, next word prediction, built-in system-wide dictionary); voice commands; and (non-English) voice recognition.

Table 20: Worldwide smartphone sales to end users by OS in 2Q15 (thousands of units)

OS	2Q15 Units	2Q15 Market Share (%)	2Q14 Units	2Q14 Market Share (%)
Android	271,010	82.2	243,484	83.8
iOs	48,086	14.6	35,345	12.2
Windows	8,198	2.5	8,095	2.8
Blackberry	1,153	0.3	2,044	0.7
Other	1,229	0.4	1,417	0.5
Total	**329,676**	**100.0**	**290,385**	**100.0**

Each of the aforementioned platforms requires a particular programming language, different development environments and programming models based on platform-specific APIs. For example, developing applications for Android requires Java, while developing applications for iOS requires Objective-C [209], and Windows requires C#/.net, which compounds the challenge of developing app(lication)s running on several of the referred Oss, because of the fragmentation around the code to develop and to maintain, the effort to debug and the knowledge required. Table 21: iOS, Windows and Android OS features comparison summarises the features provided by iOS, Windows and Android OSs [211].

Table 21: iOS, Windows and Android OS features comparison

	Virtual Machine	Programming Language	User Interface	Memory Management	IDE	Platform	Devices	App Market
Android	Dalvik VM	Java	XML files	Garbage collector	Eclipse	Multi-platform	Heterogen	Google Play Store
iOS	No	Objective-C	Cocoa Touch	Reference counting	XCode	Mac OS X	Heterogen	iTunes Apps Store
Windows Phone 7	CLR	C# and .Net	XAML files	Garbage collector	Visual Studio	Windows Vista/7	Heterogen	Windows Phone Store

The following paragraphs lay out different approaches to the concept of ***write once run anywhere*** **(WORA)** by using CP mobile development as a way to reduce the efforts invested in the maintenance and deployment processes and saves development time and effort.

5.3.2 CP Development

CP programming is seeking to fulfil the *write once run anywhere* philosophy.

CP programming is the practice of actively writing software that will work on more than one (OS) platform. It is a special case of software development, in particular for mobile apps developers, as they have to consider different aspects such as short development lifecycle, mobile device capabilities, mobility, mobile device specifications like screen sizes, the app user interface design and navigation, security and user privacy.

As a result, the developer has to consider certain issues, restrictions or challenges in order to have a successful implementation:

- Maintaining and testing CP applications is considerably more complicated, since different platforms can exhibit slightly different behaviours or subtle bugs ("write once, debug everywhere"), and they are also influenced by platform updates.
- Developers are often restricted to the lowest common subset of features or resources available on all platforms, which presents more limited computing, storage and connectivity features than fixed computers.
- Different OS platforms provide different SDKs and APIs, different hardware/software features and capabilities, user interface conventions, deployment formats and stores etc.

Table 22: Android native vs Web apps development

	Android	**Web Apps**
Application Areas		
Devices	Android devices	Any computer
Distribution	Google Play Store Local APK installation	Cloud installation/update
Required Run-time	**Infrastructure**	
Devices	Android devices Load depends on local logic	Any computer Any operating system A Web browser Load depends on complexity of content
Web Service Server	Load depends on number of users	Load depends on number of users
Development Effort		
Development	Java-Client	HTML/JavaScript/CSS Client Java Server
Network Communication	Client-side HTTP/XML-REST Interface	Server-side HTTP/XML-REST Interface
Graphical User Interface	Native Android development	Dynamic rendering of HTML/CSS/SVG content
Programming Language	Java (client-side)	Java (server-side), JavaScript (client-side)
Technologies and Standards	XML, HTTP(S)	XML, HTML 5, CSS, SVG

CP Mobile Application Types

Types of mobile applications: native, Web apps, hybrid apps and CP frameworks

Considering the way mobile applications solve these issues, they can be classified as:

- **Native apps (NA)** are created using the tools and languages provided by a certain mobile OS. Here, developing a CP app (see Chapter 5.3.2) requires multiple source trees (one for each operating system). While these applications have better performance results and best "*standardised OS user experience*", they are expensive in terms of development cost, time and sometimes licensing. They could entail more problems with regard to bug tracking and fixing.
- **Web apps (WA)** use technologies such as HTML 5, JavaScript and CSS, and they rest on the capabilities of installed Web browsers. Their installation and maintenance is very simple, and they present their results to the user performing most of the processing within the cloud or in a server. By contrast, they are not allowed to access mobile sensors, their performance is poor because of the need to interpret HTML or JavaScript, and a continuous network connection is essential. A comparison between Web apps and native Android development is shown in Table 22: Android native vs Web apps development.

- **Hybrid apps (HA)** are middle-term applications that combine Web app and native app capabilities. They are developed using Web technologies but are rendered using a native App with Web view control. The device capabilities are exposed to the hybrid app through an abstraction layer (JavaScript APIs). Although their performance is lower than native apps and their look and feel lack some of the features of the native user interfaces, they have better performance than Web apps, can access devices features, and their size depends on the need to store data locally or not.
- **Finally**, these mobile applications are developed using CP **solutions**. The app is developed once and is available to more than one platform, which reduces the total time and effort required to be invested. These kinds of applications can also be classified into two subclasses.

Firstly, they are apps with their native code being generated automatically during the implementation of the user interface. The end users interact with platform-specific native user interface components, while the application logic is implemented independently using several technologies and languages, such as Java, Ruby and XML. The main advantage of this approach is efficiency because of the native user interfaces, but the downside is the complete dependence on the software development environment. More specifically, new platform-specific features (e.g., new user interface features of a new Android version) can be available to apps only when and if supported by the development environment.

Secondly, generated apps are compiled just like a native app, and a platform-specific version of the application is created for each target platform. Generated apps achieve high overall performance because of the generated native code. It is also possible to exploit the produced native code in order to meet specific needs (e.g., in the case that a suggestion has been made from the market store to correct a deflected call), or even to adopt and extend the produced native source code to build native apps. However, in practice, utilisation of the generated native code is difficult because of its automated structure.

Table 23: Cross-platform applications comparison [210]

Decision criterion	Native Approach	Mobile Web Approach	CP Approach
Quality of UX	Excellent	Very good	Not as good as native apps
Quality of Apps	High	Medium	Medium to Low
Potential Users	Limited to a particular mobile platform users	Maximum, including smart phones, tablets and other feature phones	Large, as it reaches users of different platforms
App Development Cost	High	Medium	Medium to Low
App Security	Excellent	Depends on browser security	Not good
Supportability	Complex	Simple	Medium to complex
Ease of Updating	Complex	Simple	Medium to complex
Time-to-Market	High	Medium	Short
App Extension	Yes	Yes	Yes
Performance	Device-specific Fast rendering	Lower than the other Highly dependent on network quality	Lower than native approach
Reliability	High within the same OS device. Easy error detection.	Inconsistent behaviour due to different browsers and varying Web standards Late error detection due to dynamic typecast programming language (JavaScript)	Lower than native approach, because it depends on OS updates. Better error detection than Web approach
Flexibility	Full hardware capabilities through native development (notifications, camera, address book, geo-location, encrypted storage etc.) within the same OS.	Restricted through Web standard	Dependent on library updates to new hardware/OS capabilities

The main problem that CP solutions present is that more of them are still being researched, and most of the commercial ones are based on Web technologies. Moreover, they lack the most recent features introduced by the OSs, because the tool vendor has to support them.

CP Mobile Development Approaches

According to [211], the following CP development approaches can be discerned:

- **Compilation** uses a programme, a compiler, to transform the high-level programming language into the target lower-level language (assembly language or machine code). Depending on the kind of transformation, it can be cross-compilation, when the resulting executable can be run on a different OS from the one on which the compiler run, or trans-compilation, when the transformation is from a high-level language to another.
- Component-based approach divides the system functions into a set of separate components with defined interfaces. Each component has the same interface for all platforms, but different inner implementations based on each supported platform.
- Interpretation uses a programme, the interpreter, to translate source code instructions in real time with a dedicated engine. The developers develop the mobile applications once, and the interpreter manages to execute them on many platforms. There are three different kinds of interpreters: virtual machines (like Java Virtual Machine), which are virtual computers simulating various computer functions and on which it is possible to run an independent platform language; Web-based use, as already introduced above web technologies, they use wrappers to access device features, to the native APIs, like cameras or sensors; and run-time interpretation uses run-time, an execution environment and a layer that make the mobile app run on the native platform by transforming the application into bytecode and then, at run-time, a virtual machine supported by the mobile device executes that bytecode.
- Modelling is based on the developers' use of abstract models to describe the functions and/or the user interface of the applications. These models are then transformed into source code for different target applications. There are two sub-approaches: model-based user interface development (MB-UID), which is used to generate the UI automatically by formally describing the task, data and users for an app, and these formal models are used to guide the design generation; and model-driven development (MDD), which generates platform-specific versions on the app out of a platform-independent model, and whose abstract model is described using UML or DSL.

Table 24: Comparison of types of CP development tools

CP approaches comparison

Approach		Pros & Cons
Compilation	Cross-compiler	**Pros:** Existing code is reused by means of cross-compilation to another application; runs on different platforms; native applications. **Cons:** The mapping between the source language and the target language is difficult to achieve, so cross-compilation supports few platforms and focuses on the common elements. It also needs a code review for each platform.
	Trans-compiler	**Pros:** The legacy applications and existing source code are reused by means of trans-compilation; the generated apps are native. **Cons:** It focuses only on the common APIs in both the source and the target programming languages, and it needs regular updates to incorporate changes in the APIs of the source or the target languages.
Component-Based		**Pros**: It simplifies the support of new platforms by implementing a set of components interfacing with the new platform. **Cons**: It focuses on the common functions among all supported platforms; the developer has to learn how to use the defined component interfaces.
Interpreted	Web-based	**Pros**: Easy to learn and use, as it depends on Web technologies. **Cons:** The user interface does not have the native look and feel; applications perform worse than the native apps do.
	Virtual Machine	**Pros**: Smaller size of apps and faster downloading times from the store because all the libraries and methods needed for the app to run are stored in the VM. **Cons**: Slow execution of the application on the VM, hence the VM is not used with apps that require quick response times; the VM needs to be downloaded from the app store, which is not possible for Apple's platform (iOS).
	Run time	**Pros**: The source code is written once for all the target platforms. **Cons**: The loading performance is lower, as interpreting the source code on the device is required every time the application runs.
Modelling	MU-UID	**Pros:** Development time is saved by generating the UI code; useful in prototyping, as it allows a rapid UI development to evaluate the usability of the apps in many devices and platforms. **Cons**: It needs to focus on the similarity of user interfaces on different platforms; maintenance of the generated UI for the different platforms is difficult.
	MDD	**Pros**: The language used for modelling is an effective tool to define requirements; it helps the developers to focus on the functions of the app instead of the technical implementation issues. **Cons**: It does not support the reuse of existing native source

	code.
Cloud-Based	**Pros**: Application processing is delegated to the cloud. **Cons**: It requires a high-speed network environment; the mobile device needs an Internet connection to run the application.
Merged	**Pros**: It benefits from the strengths of each approach; it provides the developer with more features/facilities. **Cons**: It requires a lot of effort in development.

- Cloud-based approach processes application data in a cloud server instead of running the application locally. As a result, it can use cloud features like flexibility, virtualisation, security and dynamic management (see Chapter 3.1). This approach supports thin client devices, which are lightweight and potentially energy-efficient.
- Merged approach combines multiple approaches to benefit from the advantages and reduce the drawbacks of each one on its own.

Table 24 summarises the pros and cons of each of these CP frameworks, and [212] and [213] a set of criteria to compare the different CP development toolkits with each other. For a better overview, the list of criteria can be divided into infrastructure and development perspective. Table 25: Infrastructure and Development Based Criteria shows the infrastructure perspective that sums up the criteria relating to an app's lifecycle, usage, operation and functionality/functional range. This table also shows the development perspective covering all criteria directly related to the development process of the app, e.g., testing, debugging and development tools.

A very complete comparison of available CP tools is provided by the benchmarking reports at research2guidance [214]. Authors complete the previously introduced criteria with others through several statistic graphics and tables summarising the features of the 40 most popular CP frameworks based both on the available documentation and on the opinion of enterprises and 2,188 developers using them.

Table 25: Infrastructure and Development Based Criteria

Infrastructure and development criteria for comparing CP development tools

Infrastructure-Based Criteria
License and costs: This criterion examines whether the framework is distributed as free software or open source software (OSS), the license under which it is published, whether a developer is free to create commercial software, and whether there are any costs involved in support inquiries.
Supported platforms: A consideration of the number and importance of supported mobile platforms, with a special focus on whether the solution supports the platforms equally well.
Access to advanced, device-specific features: Comparison of features according to application programming interface (API) and website. Most frameworks support standard hardware (e.g., the camera), hence more advanced hardware features like near field communication (NFC) chips, and the support of multi-touch gestures is evaluated.
Long-term feasibility: Especially for smaller companies, the decision for a framework might be strategic because of the significant initial investment. Indicators for long-term feasibility are short update cycles, regular bug fixes, support of latest versions of mobile OSSs, an active community with many developers and several supporters steadily contributing to the framework's development.
Look and feel: While the general appearance of an app can be influenced during development, it does matter whether a framework inherently supports a native look and feel, or whether its user interface looks and behaves like a website. Most users seek apps that resemble native apps. Furthermore, this criterion tries to ascertain how far a framework supports the special usage philosophy and lifecycle that are inherent to an app. Apps are frequently used for a short amount of time, have to be "instantly on" and are likely to be interrupted, for example by a call. When returning to the app, a user does not want to repeat the input but wants to continue where he/she left off.
Application speed: A comparison of the application's speed when starting up and running, i.e. its responsiveness to user interaction. For evaluation, instead of measuring the performance, it assesses the subjective user experience.
Distribution: How easy is it to distribute apps created with the particular framework to consumers? One part of the answer depends on the possibility of using the app stores of mobile platforms, since clients often want to use it. However, solely relying on app stores also has disadvantages, and a framework that offers additional channels also has merits. This criterion also assesses whether updates are possible.
Development-Based Criteria
Development environment: Evaluates maturity and features of the development environment typically associated with the framework, particularly tool support (IDE, debugger, etc.) and functionalities (auto-completion, testing). The term "ease of installation" summarises the effort for setting up a fully usable development environment for a framework and a desired platform.
GUI design: This criterion covers the process of creating the graphical user interface (GUI), especially its software support. A separate WYSIWYG editor and the possibility of developing and testing the user interface without having to constantly "deploy" it to a device or an emulator are seen as beneficial.
Language needed: Find out which tool can be used with the existing development skills of your team and for which tool you need no prior programming skills.
Ease of development: This criterion sums up the quality of documentation and the learning curve. Therefore, the quality of the API and documentation is evaluated. This part of the criterion is well-fulfilled if code examples, links to similar problems, user-comments, and so forth are available. The learning curve describes developers' subjective progress during their first examination of a framework. Intuitive concepts bearing resemblance to already-known paradigms allow for fast success. This can have a significant impact on how fast new programmers can be added to a project team.
Maintainability: The indicator of lines of code (LOC) is employed to evaluate the maintainability. This indicator choice is based on the notion that an application is easier to support when it has less LOC, because training will be shorter, source code is easier to read, and so on. More sophisticated

approaches could also provide relevant indicators, but they are hard to apply, especially in the case of complex frameworks and for apps composed of different programming and markup languages.

Scalability: Scalability is based on how well larger development teams and projects can be conducted using the framework. The modularisation of frameworks and apps is highly important, as it allows increasing the number of concurrent developers.

Opportunities for further development: Determines the reusability of source code and thus assesses the risk of lock-in, which would increase if a project that started with one framework could not be transferred to another approach later on.

Speed and cost of development: Evaluates the speed of the development process and factors that hinder fast and straightforward development. Costs are not explicitly estimated, because they are taken as being dependent on the speed of development, with the assumption that one can extrapolate from differences in the salary of a JavaScript or a Java developer.

CP tools popularity.

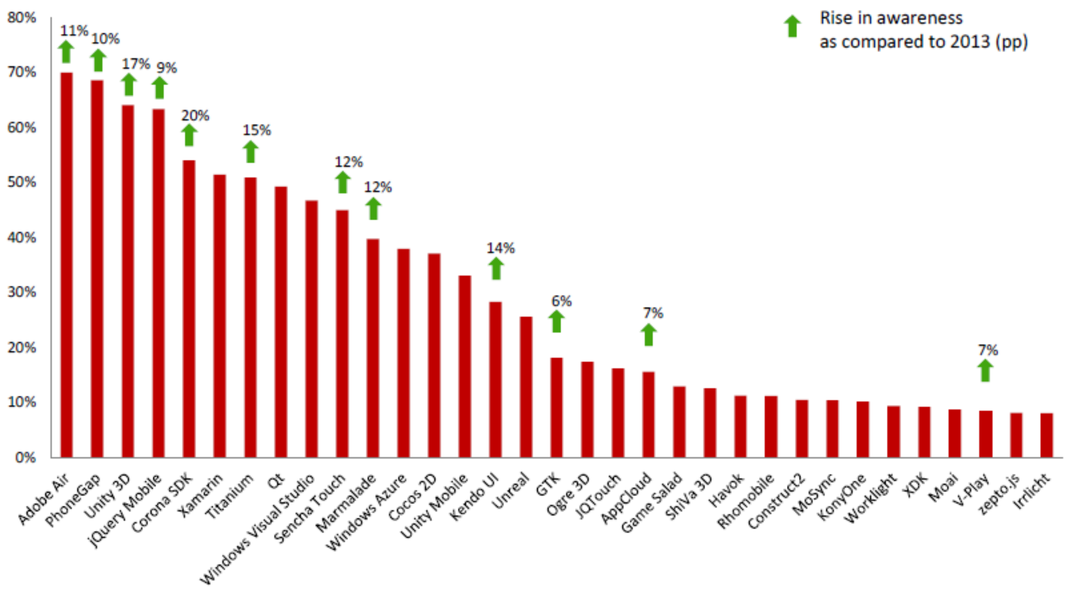


Figure 19: Developer's awareness of CP Tools [214]

Table 26 is an excerpt from the report's contents [214] showing the most defining features of the 10 most relevant CP tools in the FACTS4WORKERS scope.

Table 26: Summary of CP comparison reports

CP tools comparison

Tools	Type of Applications	OS/Platform Support	License
jQuery	WA, HA	iOS, Android, WP8, HTML5	MIT
Adobe Air	NA	iOS, Android	Free for most users
Appcelerator Titanium	NA, WA	iOS, Android, WP8	Commercial
Marmalade	NA, WA,HA	iOS, Android, WP8, HTML5	
PhoneGap	HA	iOS, Android, WP8	Apache License v 2.0
Qt	NA, WA	iOS, Android, WP8, HTML5	LGPLv2.1, LGPLv3
Unity 3D	NA	iOS, Android, WP8	Commercial
Corona	NA,HA	iOS, Android, WP8, HTML5	Commercial
Xamarin	NA, HA	iOS, Android, WP8, HTML5	Commercial
Sencha Touch	NA, HA	iOS, Android, WP8, HTML5	Commercial

5.4 Data Visualisation

Data visualization converts data sources into visual representations

Data visualisation is viewed by many disciplines as a modern equivalent to visual communication. It is a general way of talking about anything that converts data sources into a visual representation (like charts, graphs, maps, sometimes even just tables) [227]. It involves the creation and study of the visual representation of data, meaning information that has been abstracted in some schematic form, including attributes or variables for the units of information.

Table 27: Data visualisation and kinds of representations

Type of Representation	Representations
1D/Linear	Lists of data items, organised by a single feature (e.g., alphabetical order) that is not commonly ("*graphically*") visualised
2D/Planar (geospatial)	Choropleth, cartogram, dot distribution map, proportional symbol map, contour/isopleth/isarithmic map, dissymmetric map, self-organising map
3D/Volumetric	3D computer models, surface and volume rendering, computer simulations
Temporal	Timeline, time series, connected scatter plot, Gantt chart, stream graph/ThemeRiver, arc diagram, polar area/rose/circumplex chart, sankey diagram, alluvial diagram
nD/Multidimensional	Pie chart, histogram, Wordle/tag cloud, unordered bubble chart/bubble cloud, bar chart/ radial bar chart, tree map, scatter plot, bubble chart, line chart, step chart, area chart/stacked graph, heat map, parallel coordinates/parallel sets, radar/spider chart, box and whisker plot/candlestick chart, mosaic display/Marimekko chart, waterfall chart, small multiples
Tree/Hierarchical	General tree visualisation, dendrogram, radial tree, hyperbolic tree, tree map, wedge stack graph (radial hierarchy)/sunburst, icicle/partition chart
Network	Matrix, node-link diagram (link-based layout algorithm), dependency graph/circular hierarchy, hive plot, alluvial diagram, subway/tube map

Scientific, information and infographic visualisations and visual analytics form a subset of data visualisation

The following set of concepts is related to data visualisation:

- **Scientific visualisation**: In general terms, the visualisation of scientific data that have close ties to real-world objects with spatial properties. An example might be visualisations of air flow over the wing of an airplane, or 3D volumes generated from MRI scans.
- **Information visualisation**: Another broad term, it covers most statistical charts and graphs but also other visual/spatial metaphors that can be used to represent data sets that do not have inherent spatial components.

- **Infographic**: A specific sort of genre of visualisations. Infographics have become popular on the Web as a way to combine various statistics and visualisations with a narrative and, sometimes, a polemic.

Table 28: Data visualisation tools

Tool	Programming Language	License	Last Update	Type
ArborJS	JavaScript	MIT	2011	Library
Circos	JavaScript	MIT	2015	Library
Cubism	JavaScript	Apache 2.0	2012	Library
D3.js	JavaScript	BSD	2015	Library
Dance	JavaScript	MIT	2012	Library
Envisions.js	JavaScript	MIT	2015	Library
Flare	Flash	MIT	2010	Library
Google Chart Tools	JavaScript	Free to use	2015	Library
Google Fusion Tables	JavaScript, Flash	Free to use	2015	API, Desktop
JavaScript InfoVis Toolkit	JavaScript, Python	MIT	2014	Toolkit
NVD3.js	Java	Apache	2015	Bookmarked on d3.js
NodeBox	Python	GPL	2015	Desktop Application
Paper.js	JavaScript	MIT	2015	Library
Peity	JavaScript	MIT	2015	Library
Prefuse	Java	BSD	2011	Library
Processing	Processing, Java	GPL	2015	Programming Language
Processing.js	JavaScript	MIT	2015	Library
Protovis	JavaScript	BSD	2011	Library
R	R	GPL	2015	Programming Language
Raphaël	JavaScript	MIT	2015	Library
Raw	JavaScript	LGPL	2014	Web Application (on D3.js)
Rickshaw	JavaScript	MIT	2015	Library (on d3.js)
Sigma.js	JavaScript	MIT	2015	Library
TimeLine.js	JavaScript	MPL	2015	Library
Vega	JavaScript	BSD	2015	Library
Visage	JavaScript	Commercial	2015	Web Application
ZingCharts	JavaScript	Commercial	2015	Library

- **Visual analytics**: the practice of using visualisations to analyse data. In some research, visualisations can support more formal statistical tests by allowing researchers to interact with the data points directly without aggregating or summarising them. When the variables are chosen carefully, even simple scatter plots can show outliers, dense regions, bimodalities etc. In fields where the data themselves are visual (e.g., medical fields), visual analytics may actually be the primary means of analysing data. The process of analysing data through visualisation is itself studied by researchers in the visual analytics field.

Data visualisation's primary goal is to communicate information clearly and efficiently

A primary goal of data visualisation is to communicate information clearly and efficiently to users via the statistical graphics, plots, information graphics, tables, and charts selected. Effective visualisation helps users to analyse and reason about data and evidence. It makes complex data more accessible, understandable and usable. Users may have particular analytical tasks, such as making comparisons or understanding causality, and the design principle of the graphic (i.e., showing comparisons or showing causality) follows the task. Tables are generally used where users will look up a specific measure of a variable, while charts of various types are used to show patterns or relationships in the data for one or more variables.

There is a wide range of graphics types that can be used to transfer information, of which Table 27: Data visualisation and kinds of representations provides a summary. The table represents just the tip of the iceberg of all the possible visual data representations available. To help developers choose the one that best fits their requirements, several sites show examples of the charts at [228] and [229]. A full view of the available chart is provided in [230].

Several websites also provide information about tools for developing data visualisation representations, like the ones provided by [231] and [232]. From the one reviewed there, we selected the ones that are accessible via the Web. The features of these tools are presented in Table 28: Data visualisation tools.

5.5 Augmented Reality

AR definition

Azuma [3] provides a commonly accepted definition of AR as a technology that (1) combines real and virtual imagery, (2) is interactive in real time and (3) registers virtual imagery with the real world. It is this real-world element that differentiates AR from virtual reality. AR integrates and adds value to the user's interaction with the real world, as opposed to being a simulation. AR involves adding information in context to existing reality, such as statistics about a machine's performance. VR pulls the user out of his or her context by replacing the real with the virtual world [66].

User context and object of interest are essential for implementing AR

In order to achieve its objectives, an AR system should identify the context and object of interest, gather the information (data, image, etc.) to provide to the user and provide this information via the desired channel (video, audio etc). The following sections summarise the most important technologies involved in the creation of an AR application, some of which have already been introduced in this document.

5.5.1 Reality–Virtuality Continuum: AR related to other technological trends

The Reality–Virtuality Continuum (RVC) concept was first introduced by Milgram [127]. RVC is a continuous scale ranging between the completely virtual (i.e., virtuality) and the completely real (i.e., reality). The reality–virtuality continuum encompasses all possible variations and compositions of real and virtual objects [128].

Figure 20: Virtuality continuum

The area between the two extremes, where the real and the virtual are mixed, is the so-called mixed reality. This reality, in turn, is said to consist of both augmented reality – where the virtual augments the real – and augmented virtuality, where the real augments the virtual.

A wider explanation of virtual reality can be found in Annex B.

5.5.2 Augmented Reality Tracking Techniques

Table 29: Augmented reality tracking techniques briefly summarises the tracking techniques introduced in Chapter 4.8.1.

AR tracking techniques support the identification and tracking of the object of interest

Table 29: Augmented reality tracking techniques

Sensor-Based	Wi-Fi, Bluethooth, UWB, ZibBee, RFID, infrared, ultrasound
Vision-Based	Marker-based MarkerLess Natural Markers
Hybrid	A combination of different methods to improve accuracy

5.5.3 Operator Viewpoint Orientation

Operator viewpoint orientation is needed to provide location-aware and viewing direction–aware guidance

Operator Viewpoint Orientation is one of the issues that AR technologies have to resolve, because the operator's view orientation needs to be determined to provide both location-aware and viewing direction-aware guidance [71]. Here, sensors such as the **inertial measurement unit (IMU)**, a combination of accelerometers and gyroscopes, and magnetic orientation sensors (e.g., a magnetic compass) are utilised. This information can be processed to identify potential objects in the user's field to retrieve contextual information. Although the user's positional uncertainty is documented, the orientation accuracy has not been presented nor validated.

5.5.4 Interaction Techniques and User Interfaces

Tangible AR

Tangible user interfaces (TUIs) are interfaces where users can manipulate digital information with physical objects

AR bridges the real and virtual worlds so that objects in the real world can be used as AR interface elements, and their physical manipulation can provide a very intuitive way to interact with the virtual content [74]. **Tangible user interfaces (TUIs)** are interfaces where users can manipulate digital information with physical objects. Tangible interfaces are powerful because the physical objects that are used have familiar properties, physical constraints and affordances that make them easy to manipulate.

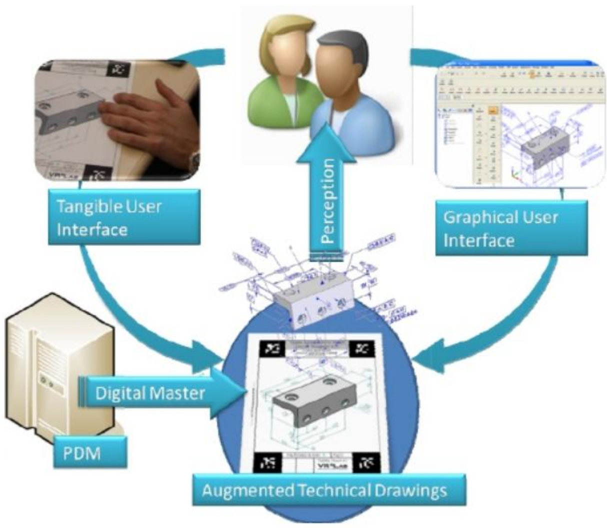


Figure 21: Tangible AR [75]

This same TUI metaphor can be applied to AR interfaces, where it is possible to combine the intuitiveness of the physical input devices with the enhanced display possibilities provided by virtual image overlays. Tangible AR interaction naturally leads to combining real-object input with gestures (see Chapter 4.5) and voice interaction (see Chapter 4.7.1), which often leads to multimodal interfaces. The overall goal of these new interaction techniques is to make it as easy to enable manipulating AR content as to interact with real objects.

Collaborative AR

Collaborative AR can support remote and co-located activities

Although single-user AR applications were studied for decades, it was not until the mid-1990s that the first collaborative AR applications were developed. Collaborative AR can support remote and co-located activities in ways that would otherwise be impossible. For co-located collaboration, AR can be used to enhance a shared physical workspace and create an interface for 3D CSCW. More recently researchers have begun exploring how mobile AR platforms can be used to enhance face-to-face collaboration. For remote collaboration, AR is able to seamlessly integrate multiple users with display devices in multiple contexts, enhancing telepresence.

5.5.5 Augmented Reality SDKs

Several SDKs are currently available on the market. In fact, the evolution of existing AR Software Development Kits (SDKs) and libraries has been so high that designers and developers today can focus on developing application logic and contents [70]. Multiple comparisons between them are available [67][68][69]. Here we provide a list of existing SDKs that have been selected based on the following criteria:

Criteria for selecting AR SDKs

- The product or library is an SDK to build AR applications, not a standalone application itself.
- The product or library exists and can be used either through a free download or a commercial license.
- The list does not include entries that represent service offerings rather than licensable software.
- The list excludes products in development that have not yet been officially released, products that were once licensable but have since been acquired by other entities who no longer offer the license publicly and products that are no longer supported by the developer or OSS community.
- The list excludes proprietary solutions that are not available to developers under any licensing model.
- Products should run on at least two mobile OSs.

Table 30 offers details about the products and a link (last visited on Sept. 30, 2015) to the company or community in charge of product development and support.

Table 30: Comparison of AR SDKs

Product	License	Platforms	Features	Comment
ALVAR	Free, Comm. SDK	Android, iOS, Windows, Flash	Marker, natural Feature	Software library for creating VR and AR applications. Developed by the VTT Technical Research Centre of Finland.
ARLab	Free, Comm. SDK	Android, iOS	GPS, IMU sensors, visual search	ARLab offers an extended portfolio of technological solutions for AR. Spain and Israel.
ARmedia	Free, Comm. SDK	Android, iOS, Windows, Flash	Marker	The ARmedia platform is a structured and modular development framework that includes different software modules arranged according to a specific architecture. The framework is independent of both the real-time tracking and rendering engines.
Arpa	Free, Commercial SDK	Android, iOS, Windows	Marker, natural feature, GPS, IMU sensors, face tracking, IR tracking, real-time rendering	Arpa Solutions is a global pioneer company developing AR products and applications through ARPA AR proprietary platform Málaga, Madrid and San Francisco
ARToolkit	OSs, Comm. SDK	Android, iOS, Linux, OSX, Windows	Marker, natural feature	ARToolworks's product line spans five OSs. Multiple CPU architectures, six languages and a myriad of APIs. In addition, the company handles the commercial licensing to a host of open source variants.
ArUco	Open Source	Linux, OSX, Windows	Marker	A minimal library for AR applications based on OpenCV
Aurasma	Free, Comm. SDK	Android, iOS	Natural feature, visual search	Owned by HP, it incorporates the company's autonomy image recognition solution.
BazAR	OSS	Linux, OSX, Windows	Natural feature	BazAR is a computer vision library based on feature points detection and matching. In particular, it is able to quickly detect and register known planar objects in images.
Beyond Reality Face	Commercial SDK	Flash	Face tracking	Beyond Reality Face Nxt is an easy-to-use face tracking solution for developers and users alike. For developers, the API is small, clear and unified for all platforms.
Catchoom	Free, Comm. SDK	Android, iOS	Visual search	Licensed by Catchoom Technologies, it offers tools to connect branded mobile application with the CraftAR service in the cloud.

DAQRI	Free, Comm. SDK	Android, iOS	Visual search, ContentAPI, natural feature	DAQRI is a vision-based AR platform that allows for maximum flexibility of both content and distribution. It offers the first enterprise-class AR solution designed to empower organisations to visualise technical assets interactively in 4D with a cloud-based data system.
HOPPALA	Free, Commercial Service	Android, iOS	Content API	Hoppala Augmentation provides an easy-to-use graphical Web interface to create AR contents with just a few mouse clicks and seamlessly publishes content to all three major AR browsers: Layar, Junaio and Wikitude.
IN2AR	Free, Comm.l SDK	Flash, iOS, Android	Natural feature	IN2AR is a CP AR engine that detects images and estimates pose using standard webcams/ mobile cameras. The pose info can be used to place 3D objects and/or videos onto the image and create AR apps or games that can be controlled by movements.
Instant Reality	Free, Comm. SDK, Closed source	Android, iOS, Linux, OSX, Windows	Marker, natural feature, GPS, IMU sensors, face tracking, visual search, content API, SLAM, tracker interface	The instant reality framework is a high-performance mixed-reality system that combines various components to provide a single and consistent interface for AR/VR developers. These components have been developed at the Fraunhofer IGD and ZGDV in close cooperation with industrial partners.
Koozyt	Comm. SDK	Android, iOS	Marker	Founded by members of Sony Computer Science Laboratories who developed the “PlaceEngine” technology in July 2007. A unique technology that connects the real and the virtual, with an emphasis on human behaviour and experience.
Layar	Free, Commercial SDK	Android, iOS	Natural feature, GPS, IMU sensors, visual search, content API	Layar enables publishers, advertisers and brands to create interactive print without hiring expensive developers or installing cumbersome software.
Mixare	OSS	Android, iOS	GPS	Mixare (mix AR Engine) is a free open source AR browser available for Android and iPhone.
Rox Odometry SDK	Free, Comm. SDK	Android, iOS, Linux, OSX, Windows	Marker, natural feature	Applications can be built with high-precision camera localisation functionalities, even in extreme conditions. Pre-recorded objects are identified and the exact position and orientation of the camera can be obtained relative to the objects in real time.
SSTT	Closed source	Android, iOS, Windows Mobile, Linux, OSX, Windows	Marker, natural feature	SSTT Bounce is a simple AR Browser. It uses a WebKit-based browser and adds fast AR NFT tracking to it.

Total Immer-sion	Free, Comm. SDK	Android, iOS, Windows, Flash	Marker, natural feature, face tracking	Total Immersion offers a commercial AR platform that integrates real-time interactive 3D graphics into a live video stream.
UART	Open Source	iOS, OSX, Windows	Marker	The Unity AR Toolkit (UART) is a set of plugins for the unity game engine that allow users to easily develop and deploy AR (AR) applications.
Vuforia	Free, Comm. SDK	Android, iOS	Marker, natural feature, visual search	Vuforia is the software platform that enables the best and most creative branded AR app experiences across the most real-world environments, giving mobile apps the power to see.
Wikitude	Free, Comm. SDK	Android, iOS, BlackBerry OS	GPS, IMU sensors, content API	Wikitude's all-in-one AR solution includes image recognition and tracking, 3D model rendering, video overlay, location-based AR and so much more.
yvision	Free, Comm. SDK	Android, iOS, Windows Mobile, OSX, Windows	Marker	YVision is a software framework that enables rapid prototyping and development of applications based on natural user interfaces. It integrates computer vision, real-time rendering, physics simulation, AR, artificial intelligence, multi-tasking and more.
ZappCode Creator	Comm. SDK	Android, iOS	Marker	The Zapcode Creator's two powerful content creation tools – the Widget Editor and Pro Editor – make it possible to create just about any kind of AR experience imaginable, including bringing a poster to life with AR-powered video or creating a fully interactive, multi-scene AR experience.

Augmented Reality Display Technologies

Screens, smart glasses and (handheld) projectors can be used to display AR

There are several ways to display AR. In Chapter 5.2.2, we introduced smart glasses. These are the most popular kind of AR displays, but AR can also be displayed using other technologies such as projection-based displays.

Projection-based display is a good option for applications that do not require several users to wear anything, and it thus provides minimal intrusiveness. A variety of projection-based display techniques has been proposed for displaying graphical information directly onto real objects or even surfaces in everyday life, from equipped rooms to handheld displays.

Handheld displays are a good alternative to HMD and HMPD systems for AR applications, particularly because they are minimally intrusive, socially acceptable, readily available and highly mobile. Currently, several types of handheld devices can be used for a mobile AR platform, including tablet PCs, UMPCS and smart phones (see Chapter 5.1).

Augmented Reality Research Challenges

Although previous tables have shown that a huge number of AR SDKs and AR visualisation devices are available in the market, some issues still remain that require special attention [70] [74]:

AR is at an early stage of development and has many challenges to face

- The first issue is related to the capability of AR applications to recognise objects in the real world and track their pose. Approaches based on the use of artificial elements – like markers – placed in the environment are very robust and work well. The main difficulty of real-time 3D tracking lies in the complexity of the scene and the motion of target objects, including the degrees of freedom of individual objects and their representation. Vision-based tracking aims to associate target locations in consecutive video frames, especially when the objects are moving quickly relative to the frame rate.
- Although marker-based tracking can enhance robustness and reduce computational requirements, it needs maintenance and often suffers from limited range and intermittent errors, because it provides location information only when markers are in sight. Therefore, marker-based methods are not scalable to handle the large-scale navigation that may be required outdoors.
- Model-based methods can capitalise on the natural features in the environment and thus extend the range of the tracking area using natural features that are relatively invariant to illuminations. Often model-based tracking methods use correspondence analysis, which in turn is supported by prediction. Model-based methods also usually require the cumbersome process of modelling, especially when creating detailed models for large a cluttered environment.
- One difficult registration problem is accurate depth perception [73]. Stereoscopic displays help, but additional problems including accommodation-convergence conflicts or low resolution and dim displays cause object to appear farther away than they should be. Correct occlusion improves some problems with depth, as that consistent registration for different eye point locations. In early video see-through systems with a parallax, users need to adapt their vision to vertically placed viewpoints.
- AR enables users to manipulate digital objects with tangible physical tools, even directly by hand. However, there are still some limitations, including (1) it is relatively difficult to tell the state of the digital data associated with physical tools, (2) the visual cues conveyed by tangible interfaces are sparse, and (3) three-dimensional imagery can be problematic in a tangible setting, as it depends on a physical display surface.

- From a human factors point of view, there are also plenty of issues to be considered. Physically, the design of the system is often cumbersome, which leads to discomfort for the user. In this sense, AR interfaces based on handheld devices are suitable, even though they have small keypads, small screens, limited resolution, small bandwidth and few computational resources. However, they do provide uniqueness and attractiveness, such as mobility, light weight and a personal gadget for social interaction, among others.
- Cognitively, the complex design of the system often makes it hard to use. Seamlessness may be more complicated to achieve among different computer platforms, display devices of different sizes and dimensionalities, and among different (local or remote) users. Usability expert Nielsen gives five quality components of usability goals, namely: learnability, efficiency, memorability, error and subjective satisfaction [77].

  Overload and over-reliance: The user interface must follow some guidelines in order not to overload the user with information while preventing the user from relying too much on the AR system and missing important cues from the environment.

Figure 22: Different HCI systems would be needed to build real, human-centred smart factories

6 Industry Readiness

The industry readiness of HCI technologies is evaluated by creating a taxonomy that confronts consumer and industrial use of HCI technologies

We now provide an evaluation of the state of the different technologies presented in the previous chapters. This evaluation is presented as a taxonomy of the HCI technologies that can be used to empower workers to carry out their daily tasks. The taxonomy is a tree, every node of which is evaluated within both the consumer world and the industrial shop-floor world. We briefly explain how we created this evaluation:

- Leaves are evaluated following modified technology readiness level (TRL) criteria. We evaluate the technology against TRL, but we also consider how easy, usable and feasible the implementation is. Consequently, we consider smart glasses to have a 9TRL level, because while they have some commercial application, it is not easy to create new systems with them. When considering the consumer or the industrial evaluation, we also consider how popular the technology is in each field.

  It must be noted that we do not evaluate a concrete product (for example, there are some commercial smart glasses). We consider each leaf as a set of products looking to the TRL level of the product, including within the node.
- Nodes are evaluated considering the truncated geometrical media of each sub-tree.

Table 31: HCI Technology readiness shows the evaluation of the technologies up to level 3 of the taxonomy. A complete version can be obtained in [239]. The following paragraphs briefly comment on the values assigned.

On the road towards implementing the Industry 4.0 vision, we provide strategies to support the coexistence of old and new technologies. Therefore we analysed existing HCI technologies and created a taxonomy. It is available on our website and updated annually:
http://facts4workers.eu/taxonomyofhcitechnologies/

6.1 HCI-Enabling Technologies

6.1.1 Conventional Technologies

Conventional HCI technologies are well-known technologies based on rugged devices and support most of the HMI interactions available on the shop floor. Although we can consider them obsolete in some respects, people know how they work. More state-of-the-art systems have included them in some way or another. Moreover, because they solve problems that are already known, we are obliged to consider them.

Conventional technologies are well-known and solve problems that are already known

Text Entry

Speech recognition is more natural and less intrusive than keyboards

Traditional text entry technologies (keyboards) are available in most devices today. However, the size of the device (or the keys), the need to use gloves and reduce intrusion in daily tasks mean that alternative approaches are required. Speech recognition seems to be the more suitable alternative, because it is less intrusive and, with the progress made with filtering algorithms, it could provide good solutions for the shop floor.

Display Devices

Although there has been a great deal of progress in the audio and haptic fields of interaction and probably because humans receive more information from visual channels, display devices are the most common way to provide information to people. While smartphone and tablet screens can be used everywhere, their use demands the attention of the users and often that they be held with at least one hand. In other words, they are not far away from the capabilities provided by a desktop or a laptop. Head-mounted displays, in particular smart glasses and (micro)-projectors, are good alternatives to using smartphone screens. The former are based on similar technologies that take advantage of today's electronic size, and they support mobility and hands-free. Projectors support hands-free, too, but they greatly depend on the lighting features of the environment.

Smart glasses and projectors require less movement and attention than screens do

Screen Positioning, Pointing and Drawing Technologies

Classical pointing technologies are present in a lot of the computers on shop floors today. They are usually attached to a fixed computer, and their use tends to be invasive with regard to the worker tasks. As it happens with keyboards, other approaches are desirable in order to provide less intrusive ways of interacting. Moreover, they are not usually supported by devices today.

Touch screens and gesture recognition contribute to more natural pointing interaction

Table 31: HCI Technology readiness

HCI Technologies	TRL Assesment: Consumer	TRL Assesment: Industrial Shop Floor
HCI Enabling Technologies	6	4
Conventional Technologies	6	5
Text Entry	7	5
Display Devices	7	5
Screen Positioning, Pointing and Drawing Technologies	6	5
Printers	7	7
Touch-sensitive Screens (Touchscreens)	9	5
Resistive Touchscreen	9	5
Capacitive Touchscreen	9	5
Infrared Touchscreen	9	5
Surface Acoustic Wave (SAW)	9	5
Image and Video Devices	9	7
2D	9	9
3D	9	6
Computer Vision	6	4
Recognition.	6	6
Motion Analysis	6	4
Scene Reconstruction	5	5
Gesture Recognition, Behavioral or Gesture Analytics	7	4
Eye Tracking	7	3
Audio Input/Output Technologies.	7	6
Speech Recognition	7	6
Text to Speech	9	6
Context Awareness Technologies	5	4
Positioning, Location and Identification Technologies.	7	5
Qualified Self	5	5
Emotion Detection, Affective Computing, Mood Recognition	4	4
Haptic Interaction	9	9
Brain Computer Interaction	3	2
HCI Systems.	6	4
Mobile Devices	9	7
Mobile Devices	9	6
Rugged Mobile Devices.	9	9
Wearable User Interfaces.	5	4
Smart Watches	7	5
Smart Glasses	6	5
Hearables	5	3
Smart Clothing	3	3
Nearables	9	9
Cross Platform (CP) Software Environments	7	5
Smart Opetating Systems.	9	5
CP Development.	7	7
Data Visualization	9	5
Scientific Visualization	9	5
Information Visualization	9	7
Infographic	9	6
Visual Analytics	9	4
Augmented Reality	5	4
Augmented Reality Tracking Techniques	5	4
Interaction Techniques and User Interfaces	6	4
Augmented Reality Display Technologies	5	5
Augmented Reality SDKs	7	5

Innovative pointing technologies provide a more natural way to interact with a computer. While touch screens are very popular because of smart phones, they are not very robust and they cannot be used easily with gloves. That is why the advances in gesture recognition or eye tracking will better support the use of ICT in industrial environments. While gesture recognition is supported by several existing smart glasses, eye tracking is still at an early stage of development.

Printers

It is difficult to visualise printers as a determinant technology of future HCI within the consumer market or on the shop floor of a factory. It is even more difficult to see it within the FACTS4WORKERS project when one of the use cases is related to the creation of a paperless factory. Nonetheless, we think they are going to be there for a while. "Old" technologies such as 2D printers or RFID printers will be there, as they form the basis of identification and location technologies today. But 3D printers (which are the base of additive production), after a time spent maturing, will probably support many maintenance works (at least to support production until the replacement piece is available).

"Old" technologies, such as 2D printers or RFID printers, are at the root of identification and location technologies today

6.1.2 Touch-sensitive Screens (Touch screens)

Touch screens are a widely used technology both in shop-floor computers today and in all kinds of tablets and smart phones. Ruggerising allows them to be used in industrial environments. However, they require users to alternate their focus of attention between their tasks and the presentation provided through the screen, which in some cases cannot be replaced. From the point of view of data introduction (the touching dimension), the use of gloves and the polish of many industrial environments would affect the feasibility of this mode of interaction. Consequently, alternative ways of interaction would be desirable in some scenarios.

Touch screens are very popular but affected by the use of gloves and environmental polish; they also require users to change their focus of attention

6.1.3 Image and Video Devices

2D cameras are devices that have been integrated into most computers. Together with the development of computer vision algorithms, they can be used to enrich user experience and, in particular, worker experience. They are used for recognise objects and places, evaluate the quality of products and detect possibly dangerous situations. While their potential is demonstrated, their main weakness is the need to have a direct overview of the scene. This issue is especially important with tablets and smart phones. Smart glasses seem to be a better alternative to provide information about the worker's point of view.

Cameras in combination with CV enrich workers' experience supporting object recognition, AR etc.

6.1.4 Computer Vision

CV is used on the shop floor under controlled conditions for quality control, security issues etc. Problems to solve are the need for training to recognise objects, loss of user point of view etc.

CV algorithms are already used on the shop floor to perform different activities related to controlling quality, avoiding problems in the interaction between humans and robots, increasing the security of workers and tracing a product and its components during the manufacturing process. Most of these approaches are based on the use of fixed cameras that are trained to perform very specific actions in a reduced set of object classes (for example, reading a barcode). Loss of direct view, occlusions and lighting conditions has a great impact on the accuracy of the algorithms. These factors have even more importance when speaking of "mobile" cameras. However, today's cameras provide of filters to improve capture images and, consequently, the result of applying CV algorithms improves day by day. By contrast, the need for specific training for each action to be performed is still required, which means an increase in cost. Finally, although today's tablets and smart phones provide very good cameras, they require users to hold them in their hands in order to take the images. Smart glasses avoid this problem, but their battery life is strongly affected by the use of the cameras.

6.1.5 Audio Input/Output Technologies

Audio communication has a low impact on the user attention, but its implementation is deeply affected by language and cultural issues

After the vision channel, the audio channel is the most relevant way for humans to exchange information. Moreover, it is less intrusive, and it has been demonstrated that it can be used while another action is being performed with a very small effect in the attention focused on it.

Different languages, accents and the environmental conditions have a great influence on the accuracy of a system implementing both speech recognition and text-to-speech conversion. Today, microphones provide several different filters that can reduce environmental noise. Several libraries provide speech recognition, and more relevant mobile operating systems also provide this kind of feature. Basic command recognition can help to improve worker interaction if it does not require complex actions.

Text-to-speech algorithms have become very popular since the automotive navigation systems (ANS) became popular about 15 years ago. They can be used to give instructions to the user without the need to watch or to look at something, in other words they allow keeping the attention on the tasks being performed. In comparison with other methods like the use of see-through capabilities at the actual state of development, speech-to-text is more comfortable for the user. It is based on the very "simple" technology that is required to interact: “just a headphone”. However, there is the problem of how to read the content. It is clear when reading a document or for an ANS (which has a reduced previously known set of instructions), but it more difficult to generalise, for example, for maintenance guiding.

6.1.6 Context-Awareness Technologies

In order to provide useful information to the workers, it is necessary to have some data about their context: where they are, what they are making, what the conditions of work are like, and how they feel. Gathering this information will support the provision of information not only to develop their daily tasks but also to improve their security and their feeling about their work. While location and identification technologies are at an advanced stage of development, this is not the case with regard to gathering information about the physiological conditions and emotions of the worker.

Location technologies are the most relevant for providing user context information

Outdoor positioning is basically based on GPS, but it is not possible to use it for indoor positioning. Indoor positioning is usually based on the deployment of a marker infrastructure either to be detected by wave sensors or cameras. Wave-based systems have the advantage of not needing a direct view of the marker, and they are not negatively affected by lighting conditions. Vision-based markers, in turn, are not affected by electrical noise, and their markers are usually cheaper than the wave-based ones. As computer vision technologies improve, they will take advantage of the use of natural markers. A special kind of markers to consider is RDFI labels, which can provide the advantages of both kinds of markers.

As important as the environmental situation of the workers is their state of health. Qualified self technologies (the information gathered by sensors worn by the workers during their working day) can contribute to improving well-being. By detecting their movements, their heart rate, etc., and by correlating them with external events, it would be possible to determine how workers react to events and, to some point, try to adapt the way they are presented to each worker profile.

QSs provide information about the worker's well-being.

Closely related to the physiological state is the emotional state of the worker. There are several different ways to detect emotions and mood: based on speech, face pose, the words that are used (in text or speech), movement etc. Emotions provide basic information about the state of the person. The problem of emotion detection is that the algorithms are computationally hard, and they are not easily generalisable.

Emotion and moods also provide valuable information about the state of the user

6.1.7 Haptic Interaction

Haptic interaction is one of the less developed modes of interaction. Even though basic solutions, such as those provided by vibrators, can be very useful to interchange information or notify about events. Since the popularisation of mobiles, they have provided this feature, which is particularly relevant if used in very noisy environments. The problem with mobiles and smart phones is that contact with the body (that is the transmission of the message) is not guaranteed. Smart watches are

Haptic is a less developed way of interaction, but vibration is a well-known option to communicate required information.

usually in direct contact with the body, and they are better candidates for implementing this kind of communication.

6.1.8 Brain–Computer Interaction

BCI is at a very early stage of development, but it must be watched in order to incorporate all its potential in the HCI on the shop floor.

6.2 HCI Systems

6.2.1 Mobile Devices

Mobile devices are well-known ways to access information anywhere

Tablets and smartphones are very popular in the consumer market, although they are not as popular in the industrial one. Rugged devices are also common but not on the manufacturing shop floor.

Mobile devices provide ubiquitous access to information, but they (usually) need to be handheld. This makes it difficult to use them to develop certain manufacturing tasks. It can be solved by using some kind of support, but it still requires a deviation of the worker's attention. The advantages of these devices are that they provide a different kind of communication (NFC, Wi-Fi, Bluetooth, 4G), sensors (NFC, GPS, IMU etc.), and they can interact with other devices (smart watches, smart glasses etc.). In many cases, they can be used as gateways to provide access to other systems to devices having lower power capabilities.

6.2.2 Wearable User Interfaces

Wearables support access data related with the user way of life but they are affected by sweat and polish and of a lack of standardisation

Wearables, in particular hearables, smart watches and smart clothing, are already providing a vast amount of data about the user's way of life. They provide information about their physiological (heart rate, body temperature etc.), activities under performance or environmental conditions (humidity, temperature etc.). This information can be used to improve the quality of the user's life, in particular, when applied to workers. However, there is a problem with their implementation at present because of the lack of contact with the body, because of sweat and polish, and because of the loss of standardisation in the supported sensors embedded in them.

Smart glasses raised a lot of attention when Google announced its Google Glass back in 2012. In the meantime, a variety of different smart glasses have been developed, many of them only as concepts or early prototypes, some of them as real products currently for sale. Although smart glasses still have not made a big impact on the consumer market, their unique technical capabilities open new possibilities for industrial applications. The chance to provide essential information through very compact visualisation combined with the benefit of hands-free usage has a high potential for the industry.

6.2.3 Cross-Platform (CP) Software Environments

One of the problems of ubiquitous HCI is that there are different platforms without a common architecture that can be used to provide interaction between the user and the systems. While Android, iOS and Windows are the most popular OSs, especially Android in the wearables field, it is not so clear for which it is better to develop an application. Native developments get better performance, but they require a development at least for each operating system; by contrast, cross-platform frameworks are at an early stage of development. Web apps are the most popular to develop cross-platform applications. Web apps require a Web browser to be installed in the device, and their performance depends on the resources that are accessible by the Web browser. Hybrid apps solve this problem by developing native modules to access devices features by the Web browser. The problem with them is that they require updates to the libraries accessing the features of the device.

The loss of common platform architecture requires tools for performing cross-platform development

6.2.4 Data Visualisation

Data visualisation is a set of tools to present information in a graphic way to users. While they are very advanced in the desktop world, they are not mobile and wearable. They need to adapt themselves to screen sizes and their interaction capabilities.

6.2.5 Augmented Reality

Augmented reality is a promise that is being fulfilled. The base technologies both for gathering information about the context and for providing the information to the user have already been developed, but content generation and the SDK is a very fragmented market that suffers the pressures of big companies. Just as an example, Metaio which provides the more popular SDK, was bought by Apple in June of 2015, but the company's subsequent disappearance has compromised many of its developments.

AR is a very promising and risky field

7 Conclusions

State of the art technologies will improve worker experience.

At present, many workstation, laptop and machine interfaces provide access to the information needed by workers on the shop floor or allow them to introduce information into the back office systems. While they can support some of the workers' needs, they require the worker to move to the place where the interfaces are. In other words, they cannot be used, for example, to support their learning in the workplace or to guide them through a task.

This document analyses the alternatives that exist to the classical WIMP interfaces in other to create a taxonomy of possible technologies that should be considered in order to fulfil the goals of FACTS4WORKERS. To this end, D2.1 analyses the theoretical background of HCI, in particular the interaction paradigms, in order to guide the technologies to be studied in this document and determine the building blocks to be created in WP2.

Once we had reviewed the available technologies, we ordered them by creating a taxonomy of available technologies. We evaluated their TRL level both in the consumer and industrial sectors, and we offered criticism of the technologies to highlight their strengths and weaknesses. The analysis of the technologies considered in the taxonomy are created from the leaves to the root. It allows us to reach some conclusions to be considered by the WP2 when a determination is made as to the technologies needed to guide the developments of the building blocks and the implementation of the requirements defined by WP1.

Mobility is a requirement of FACTS4WOR-KERS objectives

In order to support the need of ubiquity demanded by the FACTS4WORKERS goals, it is necessary to consider solutions that provide mobility. Tablets and smart phones are good candidates for this: They can communicate with other systems using Wi-Fi, mobile networks, Bluetooth, NFC and others. Also, they can get information from the user context by using both the networks and the embedded sensors, such as GPS, cameras etc. Moreover, they have enough computing power to process images, video and audio.

Wearables, gesture recognition and speech recognition can solve some of the problems of interaction on the shop floor

Although mobile devices part of the solution, they still present some problems. Firstly, they do not support total hands-free interaction, and as a result they cannot entirely support workers performing their daily tasks. Alternate ways of interaction have to be considered. Voice recognition, speech-to-text, gesture recognition and haptic interactions are already available in some wearable devices.

While smart watches and hearables can support innovative ways of audio interactions and some haptic ones, they are not efficient at providing visual information, because of the small screen or the absence of one. Most relevant smart glasses provide visual and audio communication, but they have multiple problems. The first (shared with hearables) is the need for homologation in order for them to be used on the shop floor. Some monocular and helmet-type smart glasses are being developed for use in an industrial scenario, but it is not a general feature.

A second problem, which is common to the other wearables and mobile devices, is the fragmentation of the market: both from the embedded electronic point of view and the software running on it, either the operating systems or the software to be used for creating applications.

The concentration of the OS mobile market on Android, iOS and Windows can anticipate what would happen in the wearable market

Considering the operating systems within the mobile devices market, because of the concentration produced during the past 10 years, we have limited our review to Android, iOS and Windows, which are the most popular in both the consumer and the industrial (rugged) sectors, although Android is more widespread.

Wearable devices have an even more fragmented situation: Android and iOS are the most popular, but many available devices provide a "proprietary" OS derived from Android. This is the bad news. The good news is that most of them are compatible with most popular mobile operating systems, which allows the smart devices to interact with the rest of the world using different devices. More bad news is that most of them provide a proprietary SDK, which makes the development of applications difficult because of the need to develop an application for each device.

Even the present development of applications requires a cross-platform strategy

The last problem is common when dealing with different platforms where the systems need to be deployed and the hardware and operating systems do not make it possible. It is known as cross-platform development, and over the years several solutions have been presented from native developments (better performance having the cost or more resources to develop for each platform – here considered as hardware + software) to cross-platform SDKs, which are promising but are at an early stage of development, which means selecting one is fraught with risk. As with other things, middle-way solutions can be the best choice. Web development or hybrid development should have to be considered when implementing HCI in the way FACTS4WORKERS wants to. Web development depends on the use of a state-of-the-art Web browser, which wearables do not always provide, and it has some performance and device feature permission problems. It is solved by hybrid developments by creating libraries to access device features and implementing the mobile solution on it. A cloud-based approach can also be considered to resolve performance problems, for example, when computer power is needed to process an image or a sound, but it requires ensuring device connectivity and broadband in order to guarantee a user (worker) experience that is good enough.

AR is a particular case of [CP] SDKs where the recent disappearance of a reference company compromises many SMEs in the future

A particular set of SDKs are related to the development of AR applications. As with the other issues previously analysed in this document, it is a very fragmented and dynamic field: There are many commercial and OSS development frameworks, several of them supporting cross-platform development, but the field is very dynamic, and making an *a priori* good selection is difficult. As an example, until June 2015, metaio was the choice of many companies to develop their AR applications because of the feasibility and the support they provide. But Apple bought it in June, and the company and associated software literally disappeared. A similar situation played out with Vuforia. It also produced licensed commercial software and was bought by PTC in the middle of October 2015 from Qualcomm but it seems to be still available. An alternative way to proceed with AR applications is the creation of building blocks based on the development available on OSS computer vision technologies such as OpenCV one. Here the problem is the cost of development (because of the low-level building blocks), their adaptation to the particular devices and the creation of the content to be shown.

The very dynamic market could also lead to project success because of the selection of the last available gadget

Another example of the dynamism of the market is the continuous appearance of different devices. These include Cicret [236]; ProGlove [237] and Rufus Cuff [238]. Dealing with these and future devices highlights their strengths and weakness, including them in existing branches of the taxonomy and creating new ones are things that we should be considered in future versions of this document.

References

[1] BEAUDOUIN-LAFON, Michel. Designing interaction not interfaces. *Proceedings of the working conference on advanced visual interfaces*. ACM, 2004. p. 15-22.
[2] HARRISON, Steve; TATAR, Deborah; SENGERS, Phoebe. The three paradigms of HCI. *alt. chi*, 2007.
[3] KUHN, Thomas S. The structure of scientific revolutions. *Chicago/London*, 1970.
[4] ALTMAN, Ross; GHUBRIL, Adib Carl; VELOSA, Alfonso. *Best Practices for Choosing Human-Machine Interface Technologies*. GARTNER, 2013-10-28.
[5] HEWETT, B.; CARD, C.; GASEN; M.; PERLMAN, S. *ACM SIGCHI Curricula for Human-Computer Interaction*. ACM SIGCHI. http://old.sigchi.org/cdg/cdg2.html#2_1, Retrieved 18 March 2015.
[6] MONTUSCHI, Paolo, et al. Human-Computer Interaction: Present and Future Trends. *Computing Now*, 2014, vol. 7, no 9. IEEE Computer Society [online]; http://www.computer.org/web/computingnow/archive/september2014.
[7] KARRAY, Fakhreddine, et al. Human-computer interaction: Overview on state of the art. 2008.
[8] ISO-9241. Usability definition. http://www.w3.org/2002/Talks/0104-usabilityprocess/slide3-0.html . Retrieved 19 March 2015.
[9] MORVILLE,P. *User Experience Design*. http://semanticstudios.com/user_experience_design/ . Retrieved 19 March 2015.
[10] *User Experience Definitions* All About Use Experience Web Site. http://www.allaboutux.org/ux-definitions . Retrieved 20 March 2015.
[11] JACOB, Robert JK. *Human-computer interaction: input devices*. ACM Computing Surveys (CSUR), 1996, vol. 28, no 1, p. 177-179.
[12] WHITWORTH, Brian, et al. Fitts's Law.
[13] HINCKLEY, K., JACOOB, R., WARE, C., *Input/Output Devices and Interaction Techniques*. Chapter 20 in The Computer Science Handbook, Second Edition, ed. by A.B. Tucker, Chapman and Hall / CRC Press, Boca Raton, FL, Nov. 29, 2004, pp. 20.1 – 20.32.
[14] MacKENZIE, I. S. (1995). *Input devices and interaction techniques for advanced computing*. In W. Barfield, & T. A. Furness III (Eds.), Virtual environments and advanced interface design, pp. 437-470. Oxford, UK: Oxford University Press.
[15] CHAPANIS, A. (1965). *Man-machine engineering*. Belmont, CA: Wadsworth.
[16] BOLT, R. (1980). *Put-that-there: Voice and gesture at the graphics interface*. Computer Graphics, 14(3), 262-270
[17] BOLT, R., & HERRANZ, E. (in press). *Two-handed gesture in multi-modal natural dialog*. Proceedings of the ACM SIGGRAPH and SIGCHI Symposium on User Interface Software and Technology. New York: ACM.
[18] THORISSON, K. R., KOONS, D. B., & Bolt, R. A. (1992). *Multi-modal natural dialogue*. Proceedings of the CHI '92 Conference on Human Factors in Computing Systems, 653-654. New York: ACM.
[19] CARD, Stuart K.; MORAN, Thomas P.; NEWELL, Allen. The model human processor: An engineering model of human performance. Handbook of Human Perception, 1986, vol. 2.
[20] *The Gestalt Principles*. http://www-faculty.cs.uiuc.edu/~bpbailey/teaching/2004-Fall/cs498/

[21] DIX, Alan. *Human-computer interaction*. Springer US, 2009.
[22] http://graphicdesign.spokanefalls.edu/tutorials/process/gestaltprinciples/gestaltprinc.htm. Retrieved on 08/05/2015.
[23] TULVING, Endel; THOMSON, Donald M. *Encoding specificity and retrieval processes in episodic memory.* Psychological review, 1973, vol. 80, no 5, p. 352
[24] BAILEY, Brian P., *Computer Science 498bpb, Psychology of HCI*. http://searchfiletype.com/fsearch/3/3/1/5/modelHumanProcessor_331521.pdf. Last visited on 08/05/2015.
[25] WIKIPEDIA. *Ergonomic Keyboard*. http://en.WIKIPEDIA.org/wiki/Ergonomic_keyboard Last visited 08/05/2015.
[26] PLAMONDON, Réjean; SRIHARI, Sargur N. *Online and off-line handwriting recognition: a comprehensive survey.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2000, vol. 22, no 1, p. 63-84.
[27] NARANG, Shreya; GUPTA, Ms Divya. *Speech Feature Extraction Techniques: A Review*. 2015.
[28] SCGNEIDEMAN, R., Deng, L., & Sejnoha, V. (2015). *Accuracy, Apps Advance Speech Recognition*. IEEE SIGNAL PROCESSING MAGAZINE, 1053(5888/15).
[29] GOEL, Meenal; SINGH, Sukhwinder. *Speech Recognition Techniques-A Review*.
[30] WIGAS, G. and NAVDEEP, S., "Literature Review on Automatic Speech Recognition", International Journal of Computer Applications vol.41– no.8, pp. 42-50, March 2012.
[31] WIKIPEDIA. *Light Pen*. http://en.WIKIPEDIA.org/wiki/Light_pen . Last visited 11/05/2015.
[32] WIKIPEDIA. *Digital Pen*. http://en.WIKIPEDIA.org/wiki/Digital_pen. Last visited 11/05/2015.
[33] WIKIPEDIA. *Graphic Tablet*. http://en.WIKIPEDIA.org/wiki/Graphics_tablet. Last visited 11/05/2015.
[34] WIKIPEDIA. *Touchpad*. http://en.WIKIPEDIA.org/wiki/Touchpad. Last visited 11/05/2015.
[35] WIKIPEDIA. *Trackball*. http://en.WIKIPEDIA.org/wiki/Trackball. Last visited 11/05/2015.
[36] WIKIPEDIA. *Joystick*. http://en.WIKIPEDIA.org/wiki/Joystick. Last visited 11/05/2015.
[37] WIKIPEDIA. *Analog Stick*. http://en.WIKIPEDIA.org/wiki/Analog_stick . Last visited 11/05/2015.
[38] WIKIPEDIA. TouchScreen. http://en.WIKIPEDIA.org/wiki/Touchscreen. Last visited 11/05/2015.
[39] BHALLA, Mudit Ratana; BHALLA, Anand Vardhan. *Comparative study of various touchscreen technologies*. International Journal of Computer Applications, 2010, vol. 6, no 8, p. 12-18.
[40] MEYYARASU, N.; DALTON, G.; ABINAYA, S. *A REVIEW ON TOUCH SENSOR SCREEN SYSTEM*. 2015.
[41] MORIMOTO, Carlos H.; MIMICA, Marcio RM. *Eye gaze tracking techniques for interactive applications*. Computer Vision and Image Understanding, 2005, vol. 98, no 1, p. 4-24.
[42] MAJARANTA, Päivi; BULLING, Andreas. Eye Tracking and Eye-Based Human–Computer Interaction. En Advances in Physiological Computing. Springer London, 2014. p. 39-65.
[43] WIKIPEDIA. Gesture Recognition. http://en.WIKIPEDIA.org/wiki/Gesture_recognition . Last Visited 14/05/2015.

[44] MITRA, Sushmita; ACHARYA, Tinku. Gesture recognition: A survey. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, 2007, vol. 37, no 3, p. 311-324.

[45] RAUTARAY, Siddharth S.; AGRAWAL, Anupam. Vision based hand gesture recognition for human computer interaction: a survey. Artificial Intelligence Review, 2015, vol. 43, no 1, p. 1-54.

[46] PANDEY, Pratibha; JAIN, Vinay. Hand Gesture Recognition for Sign Language Recognition: A Review. Hand, 2015, vol. 4, no 3.

[47] LANGE, Markus. Comparing Technologies for Gestural Interaction in the Car. Human-Computer Interaction in the Car, 2014, p. 2.

[48] KHAN, Rafiqul Zaman; IBRAHEEM, Noor Adnan. HAND GESTURE RECOGNITION: A Literature. 2012.

[49] IBRAHEEM, Noor Adnan; KHAN, RafiqulZaman. Survey on various gesture recognition technologies and techniques. International journal of computer applications, 2012, vol. 50, no 7, p. 38-44.

[50] KHAN, Rafiqul Zaman; IBRAHEEM, Noor Adnan. Comparative study of hand gesture recognition system. En Proc. of International Conference of Advanced Computer Science & Information Technology in Computer Science & Information Technology (CS & IT). 2012. p. 203-213.

[51] KOYUNCU, Hakan; YANG, Shuang Hua. A survey of indoor positioning and object locating systems. IJCSNS International Journal of Computer Science and Network Security, 2010, vol. 10, no 5, p. 121-128.

[52] KARSAI, Szabolcs; TÓTH, Zsolt. Comparison of WiFi-based indoor positioning techniques.

[53] STOJANOVIĆ, Dragan; STOJANOVIĆ, Natalija. INDOOR LOCALIZATION AND TRACKING: METHODS, TECHNOLOGIES AND RESEARCH CHALLENGES. Facta Universitatis, Series: Automatic Control and Robotics, 2014, vol. 13, no 1, p. 57-72.

[54] In-Location Alliance. Home Page. http://inlocationalliance.org/ . Last visited on 19-05-2015.

[55] LEE, Jin-Shyan; SU, Yu-Wei; SHEN, Chung-Chou. A comparative study of wireless protocols: Bluetooth, UWB, ZigBee, and Wi-Fi. En Industrial Electronics Society, 2007. IECON 2007. 33rd Annual Conference of the IEEE. IEEE, 2007. p. 46-51.

[56] POTHUGANTI, Karunakar; CHITNENI, Anusha. A Comparative Study of Wireless Protocols: Bluetooth, UWB, ZigBee, and Wi-Fi.

[57] NI, Lionel M., et al. LANDMARC: indoor location sensing using active RFID. Wireless networks, 2004, vol. 10, no 6, p. 701-710.

[58] HIGHTOWER, J., BORRIELLO, G., "SpotON: An indoor 3D location sensing technology based on RF signal strength," UW CSE 00 - 02 - 02, University of Washington, Department of Computer Science and Engineering, Seattle, WA, 2000

[59] FENG, H. U. Emerging Techniques in Vision-based Indoor Localization. 2015. Tesis Doctoral. The City University of New York.

[60] DEAK, Gabriel; CURRAN, Kevin; CONDELL, Joan. A survey of active and passive indoor localisation systems. Computer Communications, 2012, vol. 35, no 16, p. 1939-1954.

[61] WIKIPEDIA. Computer Vision Definition. http://en.WIKIPEDIA.org/wiki/Computer_vision . Last Visited 22-05-2015.

[62] ISHEAWY, Najib Ali Mohamed; HASAN, Habibul. Optical Character Recognition (OCR) System.

[63] WIKIPEDIA. Image Processing. http://en.WIKIPEDIA.org/wiki/Image_processing . Last visited 25-05-2015.
[64] Machine Vision Home Page. http://www.machinevision.co.uk/ Last visited 26-05-2015.
[65] MULLONI, Alessandro, et al. Indoor positioning and navigation with camera phones. Pervasive Computing, IEEE, 2009, vol. 8, no 2, p. 22-31.
[66] ZETTIN, Minda, 4 Misconceptions About Augmented Reality. http://www.oracle.com/us/corporate/profit/big-ideas/012715-bi-tl-augmented-sidebar-2415737.html . Last visited 26-05-2015.
[67] PADZENSKY, Ron. Augmented Reality SDKs. http://augmera.com/?p=461 .Last visited 01-06-2015.
[68] REITRMAYR, Gerhard. Augmented Reality SDKs. http://www.icg.tugraz.at/Members/gerhard/augmented-reality-sdks
[69] AMIN, Dhiraj; GOVILKAR, Sharvari. COMPARATIVE STUDY OF AUGMENTED REALITY SDK'S.
[70] LAMBERTI, Fabrizio, et al. Challenges, Opportunities and Future Trends of Emerging Techniques for Augmented Reality-based Maintenance.//
[71] KOCH, Christian, et al. Natural markers for augmented reality-based indoor navigation and facility maintenance. Automation in Construction, 2014, vol. 48, p. 18-30.
[72] AZUMA, R. T. (1997). A survey of augmented reality. Presence-Teleoperators and Virtual Environments, 6(4), 355–385.
[73] VAN KREVELEN, D. W. F.; POELMAN, R. A survey of augmented reality technologies, applications and limitations. International Journal of Virtual Reality, 2010, vol. 9, no 2, p. 1.
[74] ZHOU, Feng; DUH, Henry Been-Lirn; BILLINGHURST, Mark. Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR. En Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality. IEEE Computer Society, 2008. p. 193-202.
[75] FIORENTINO, M., Tangible digital master for product lifecycle management in augmented reality. International Journal on Interactive Design Manufacturing (IJIDEM).
[76] BIMBER, O., GRUNDHOFER, A., WETSTEIN, G and KNODEL, S. Consistent illumination within optical see-through augmented environments. In ISMAR '03, pp. 198-207, 2003.
[77] NIELSEN, J., (1994). Usability Engineering (1st ed.). San Francisco: Morgan Kaufmann.
[78] GUTTAG, K. HMD – A huge number of options. http://www.kguttag.com/category/near-eye-augmented-reality/ Last Visited 03/06/2015.
[79] PADZENSKY, Ron. Survey of Head Monted Displays. http://augmera.com/?p=504. Last visited on 05/06/2015.
[80] WIKIPEDIA. Cross-platform definition. http://en.WIKIPEDIA.org/wiki/Cross-platform. Last visited on 05/06/2015.
[81] DE BABO MARTINS, Francisco. Visual-Inertial based autonomous navigation of an Unmanned Aerial Vehicle in GPS-Denied environments. 2015.
[82] ZEIS, Adam. Ultimate guide to buying the best smartwatch. http://www.connectedly.com/smartwatch-buyers-guide#content. Last visited on 13/07/2015.
[83] RAWASSIZADEH, Reza; PRICE, Blaine A.; PETRE, Marian. Wearables: has the age of smartwatches finally arrived? Communications of the ACM, 2014, vol. 58, no 1, p. 45-47.
[84] ISACSON, Dan. Application development for smartwatches. June, 2015.

[85] BOUHNICK, Gil. The fragmented world of wearable operating systems. February, 2015.
[86] KANNON, Y., What Operating Systems Do Wearable Devices Run On? http://www.makeuseof.com/tag/what-operating-systems-do-wearable-devices-run-on/ Last visited on 15/07/2015.
[87] SONEIRA,R.M. Smart Watch Display Technology Shoot-Out. http://www.disp/laymate.com/Smart_Watch_ShootOut_1.htm. Last visited on 21/11/2015.
[88] 2015 Smartwatch Specs Comparison Chart. http://smartwatchme.s3.amazonaws.com/1369262f9a93c6b302d063082c296c9176822baa2806.pdf. Last visited on 16/07/2015.
[89] CHENNAMMA, H. R.; YUAN, Xiaohui. A survey on eye-gaze tracking techniques.arXiv preprint arXiv:1312. 6410, 2013.
[90] KIEFER, Peter; STRAUB, Florian; RAUBAL, Martin. Towards location-aware mobile eye tracking. En Proceedings of the Symposium on Eye Tracking Research and Applications. ACM, 2012. p. 313-316.
[91] BOHME, Martin, et al. Remote eye tracking: State of the art and directions for future development. En Proc. of the 2006 Conference on Communication by Gaze Interaction (COGAIN). 2006. p. 12-17.
[92] BULLING, Andreas; GELLERSEN, Hans. Toward mobile eye-based human-computer interaction. Pervasive Computing, IEEE, 2010, vol. 9, no 4, p. 8-12.
[93] BULLING, Andreas; DUCHOWSKI, Andrew T.; MAJARANTA, Päivi. PETMEI 2011: the 1st international workshop on pervasive eye tracking and mobile eye-based interaction. En Proceedings of the 13th international conference on Ubiquitous computing. ACM, 2011. p. 627-628.
[94] JOHNSON, D.; MALTZ, D. Mobile computing. 1996.
[95] SATYANARAYANAN, Mahadev. Fundamental challenges in mobile computing. En Proceedings of the fifteenth annual ACM symposium on Principles of distributed computing. ACM, 1996. p. 1-7.
[96] WIKIPEDIA. Mobile Computing. https://en.WIKIPEDIA.org/wiki/Mobile_computing. Last visited 18-09-2015
[97] SCHOOLER, Eve M. Conferencing and collaborative computing. Multimedia Systems, 1996, vol. 4, no 5, p. 210-225.
[98] SCHMIDT, Albrecht; LAUFF, Markus; BEIGL, Michael. Handheld cscw. En Workshop on Handheld CSCS at the 1998 ACM conference on Computer-Supported Cooperative Work. 1998.
[99] WIKIPEDIA. Computer-supported cooperative work. https://en.WIKIPEDIA.org/wiki/Computer-supported_cooperative_work. Last visited 18-09-2015.
[100] RICHARDSON, Tristan, et al. Virtual network computing. Internet Computing, IEEE, 1998, vol. 2, no 1, p. 33-38.
[101] ZHANG, Qi; CHENG, Lu; BOUTABA, Raouf. Cloud computing: state-of-the-art and research challenges. Journal of internet services and applications, 2010, vol. 1, no 1, p. 7-18.
[102] PADHY, Rabi Prasad; PATRA, Manas Ranjan. Evolution of cloud computing and enabling technologies. International Journal of Cloud Computing and Services Science (IJ-CLOSER), 2012, vol. 1, no 4, p. 182-198.

[103] WEISER, Mark. Some computer science issues in ubiquitous computing. Communications of the ACM, 1993, vol. 36, no 7, p. 75-84.

[104] SALBER, Daniel; DEY, Anind K.; ABOWD, Gregory D. Ubiquitous computing: Defining an HCI research agenda for an emerging interaction paradigm. 1998.

[105] WIKIPEDIA. Multimodal Interaction. https://en.WIKIPEDIA.org/wiki/Multimodal_interaction. Last visited on 21/09/2015.

[106] WIKIPEDIA. Wearable Computers. https://en.WIKIPEDIA.org/wiki/Wearable_computer. Las visited on 21/09/2015.

[107] JAMESON, Anthony. Adaptive interfaces and agents. Human-Computer Interaction: Design Issues, Solutions, and Applications, 2009, vol. 105.

[108] MCCULLOUGH, Malcolm. Digital ground: architecture, pervasive computing, and environmental knowing. The MIT Press, 2005.

[109] BATYA FRIEDMAN. Human values and the design of computer technology. Cambridge University Press, 1997.

[110] NORMAN, Donald A. Emotional design: Why we love (or hate) everyday things. Basic books, 2004.

[111] BORENSTEIN, Nathanial S. Computational mail as network infrastructure for computer-supported cooperative work. En Proceedings of the 1992 ACM conference on Computer-supported cooperative work. ACM, 1992. p. 67-74.

[112] BAECKER, Ronald M. (ed.). Readings in Human-Computer Interaction: toward the year 2000. Morgan Kaufmann, 2014.

[113] WIKIPEDIA. Speech Recognition. https://en.wikipedia.org/wiki/Speech_recognition. Last Visited on 29/10/2015

[114] GRIFONI, Patrizia, et al. MIS: Multimodal Interaction Services in a cloud perspective. Journal of Next Generation Information Technology, 2014, vol. 5, no 4, p. 1.

[115] KETTEBEKOV, Sanshzar; SHARMA, Rajeev. Toward natural gesture/speech control of a large display. Engineering for human-computer interaction. Springer Berlin Heidelberg, 2001. p. 221-234.

[116] VASSILIOU, Marius S., et al. Integrated multimodal human-computer interface and Augmented Reality for interactive display applications. En *AeroSense 2000*. International Society for Optics and Photonics, 2000. p. 106-115.

[117] VITENSE, Holly S.; JACKO, Julie A.; EMERY, V. Kathlene. Multimodal feedback: establishing a performance baseline for improved access by individuals with visual impairments. En Proceedings of the fifth international ACM conference on Assistive technologies. ACM, 2002. p. 49-56.

[118] OVIATT, Sharon. Multimodal interfaces. The human-computer interaction handbook: Fundamentals, evolving technologies and emerging applications, 2003, vol. 14, p. 286-304.

[119] BROOKS, Anthony; PETERSSON, Eva. SoundScapes: non-formal learning potentials from interactive VEs. En ACM SIGGRAPH 2007 educators program. ACM, 2007. p. 18.

[120] FITTS, Paul M. The information capacity of the human motor system in controlling the amplitude of movement. Journal of experimental psychology, 1954, vol. 47, no 6, p. 381.

[121] HICK, William E. On the rate of gain of information. Quarterly Journal of Experimental Psychology, 1952, vol. 4, no 1, p. 11-26.

[122] WIKIPEDIA. Output Devices. https://en.WIKIPEDIA.org/wiki/Output_device. Last visited on 07/10/2015.

[123] Lemley, Linda. "Chapter 6: Output". Discovering Computers. University of West Florida. Last visited on 08/10/2015.
[124] WIKIPEDIA. Display Devices. https://en.WIKIPEDIA.org/wiki/Display_device. Last visited on 08/10/2015.
[125] Accommodations For Vision Disabilities. Energy.gov. Office of the Chief information Officer. Last visited on 03/06/2012.
[126] WIKIPEDIA. Gamut. https://en.WIKIPEDIA.org/wiki/Gamut. Last visited on 08/10/2015.
[127] MILGRAM, Paul, et al. Augmented reality: A class of displays on the reality-virtuality continuum. En Photonics for Industrial Applications. International Society for Optics and Photonics, 1995. p. 282-292.
[128] WIKIPEDIA. Reality–virtuality continuum. https://en.WIKIPEDIA.org/wiki/Reality%E2%80%93virtuality_continuum. Last visited on 14/10/2015
[129] TYSON, J., and CARMACK, C., "How Computer Monitors Work" 16 June 2000. HowStuffWorks.com. <http://computer.howstuffworks.com/monitor.htm> Last visited on 14/10/2015.
[130] FERNÁNDEZ, María Rodríguez; CASANOVA, Eduardo Zalama; ALONSO, Ignacio González. Review of Display Technologies Focusing on Power Consumption. Sustainability, 2015, vol. 7, no 8, p. 10854-10875.
[131] E-Paper Display (EPD) Market Analysis, Market Size, Application Analysis, Regional Outlook, Competitive Strategies And Forecasts, 2014 To 2020. http://www.grandviewresearch.com/industry-analysis/e-paper-display-epd-market. Last visited on 15/10/2015.
[132] WIKIPEDIA. Electronic Paper. https://en.WIKIPEDIA.org/wiki/Electronic_paper. Last visited on 15/10/2015
[133] Future Electronics. What is OLED?http://www.futureelectronics.com/en/display-solutions/oled.aspx . Last visited on 15/10/2015.
[134] WIKIPEDIA. OLED. https://en.WIKIPEDIA.org/wiki/OLED. Last visited on 15/10/2015.
[135] WIKIPEDIA. Digital Light Processing. https://en.WIKIPEDIA.org/wiki/Digital_Light_Processing. Last visited on 15/10/2015.
[136] WIKIPEDIA. Handheld Projectors. https://en.WIKIPEDIA.org/wiki/Handheld_projector. Last visited on 15/10/2015.
[137] Buyer's Guide to Business Projectors: Key Features. http://www.projectorcentral.com/buyers6.htm. Last visited on 15/10/2015.
[138] CHRISTENSEN, Henrik Bærbak. Cross Platform Mobile Development. 2015.
[139] WIKIPEDIA. Comparison of Mobile Operating systems. https://en.WIKIPEDIA.org/wiki/Comparison_of_mobile_operating_systems . Last visited on 15/10/2015.
[140] WIKIPEDIA. Mobile Operative Systems. https://en.WIKIPEDIA.org/wiki/Mobile_operating_system. Last visited on 15/10/2015.
[141] GARTNER. GARTNER Says Worldwide Smartphone Sales Recorded Slowest Growth Rate Since 2013. http://www.GARTNER.com/newsroom/id/3115517. Last visited on 15/10/2015.

[142] WIKIPEDIA. 3D Printing. https://en.WIKIPEDIA.org/wiki/3D_printing, Last visited on 23/10/2015.
[143] RFID Printers. http://www.barcoding.com/rfid/rfid-printers.shtm. Last visited on 23/10/2015.
[144] BAGULEY, R. 3D Printer Buyer's Guide 2015. http://www.tomsguide.com/us/3d-printer-buyers-guide,news-17651.html. Last visited on 23/10/2015.
[145] Esterline. Touchscreen Technology Comparison. https://www.esterline.com/Portals/13/Literature/Literature/DS_TouchscreenTechnology_Letter.pdf . Last visited on 23/10/2015.
[146] WIKIPEDIA. Computer Vision. https://en.WIKIPEDIA.org/wiki/Computer_vision. Last visited on 23/10/2015.
[147] WIKIPEDIA. openCV. https://en.WIKIPEDIA.org/wiki/OpenCV. Last visited on 26/10/2015.
[148] openCV Home Page. http://opencv.org/. Last visited on 26/10/2015.
[149] Wii Remote. http://www.nintendo.com/wiiu/accessories. Last Visited on 26/10/2015.
[150] MYO home page. https://www.myo.com/ . Last Visited on 26/10/2015.
[151] SCOOP POINTER REMOTE. http://hillcrestlabs.com/products/hardware/scoop-pointer-remote/. Last visited on 26/10/2015.
[152] Tobii Pro Glasses 2. http://www.tobiipro.com/product-listing/tobii-pro-glasses-2/ Last visited on 26/10/2015.
[153] WIKIPEDIA. Mobile Device. https://en.WIKIPEDIA.org/wiki/Mobile_device Last visited on 27/10/2015.
[154] ROUSE, M. rugged IT.http://whatis.techtarget.com/definition/rugged-IT . Last visited on 27/10/2015.
[155] WIKIPEDIA. Ingress Protection Code. https://en.WIKIPEDIA.org/wiki/IP_Code. Last visited on 27/10/2015.
[156] Scandit. Rugged Smartphones in the Enterprise. 2015 Buyers Guide. http://www.scandit.com/wp-content/uploads/2015/05/Rugged-Smartphones-2015-eBook1.pdf
[157] USA, DoD. MIL-STD-810F http://www.atec.army.mil/publications/Mil-Std-810G/Mil-Std-810G.pdf.
[158] Spectout. Compare Rugged Smartphones http://smartphones.specout.com/d/p/Rugged-Smartphone. Last visited on 27/10/2015.
[159] DPReview. Camera feature review. http://www.dpreview.com/products/compare/side-by-side?products=canon_ixus115hs&products=canon_ixus500hs&products=canon_eos6d Last visited on 27/10/2015.
[160] HENRY, Peter, et al. RGB-D mapping: Using depth cameras for dense 3D modeling of indoor environments. En In the 12th International Symposium on Experimental Robotics (ISER. 2010).
[161] LI, Larry. Time-of-Flight Camera–An Introduction. Technical White Paper, May, 2014.
[162] Extend3D. Werklicht Pro. http://www.extend3d.de/en/werklichtpro.php Last visited on 27/10/2015.
[163] Whatis.com. Hearables. http://whatis.techtarget.com/definition/hearables . Last visited on 28/10/2015.

[164] WIKIPEDIA. Hearables. https://en.WIKIPEDIA.org/wiki/Hearables Last visited on 28/10/2015.
[165] Extreme Tech. 'Hearables' could be the next wearables — and with good reason. http://www.extremetech.com/mobile/205675-hearables-could-be-the-next-wearables-and-with-good-reason. Last visited on 28/10/2015.
[166] PROFIS, S. Do wristband heart trackers actually work? A checkup. CNET, 22/05/2014. http://www.cnet.com/news/how-accurate-are-wristband-heart-rate-monitors/. Last visited on 28/10/2015.
[167] Tourism Think Tank. Wearables, Hearables and Nearables. http://thinkdigital.travel/opinion/marketing-and-selling-with-wearables-hearables-and-nearables Last visited on 28/10/2015.
[168] Elbee Home Page. https://getelbee.com/ Last visited on 28/10/2015.
[169] Bragi Home Page. The Dash. http://www.bragi.com/ . Last visited on 28/10/2015.
[170] Waberly Labs Home Page. http://www.waverlylabs.com/. Last visited on 28/10/2015.
[171] Motorola. Moto Hint. Home Page. http://www.motorola.com/us/accessories/talk/bluetooth-headsets/moto-hint/moto-hint-pdp.html. Last visited on 28/10/2015.
[172] WIKIPEDIA. Nearables. https://en.WIKIPEDIA.org/wiki/Nearables. Last visited on 28/10/2015.
[173] Estimote. iBeacons. http://developer.estimote.com/ibeacon/ . Last visited on 28/10/2015.
[174] Apple. iBeacons for developers. https://developer.apple.com/ibeacon/ Last visited on 28/10/2015.
[175] Google. Eddystone. https://developers.google.com/beacons/ Last visited on 28/10/2015.
[176] Estimote. What's Eddystone?. http://developer.estimote.com/eddystone/ Last visited on 28/10/2015.
[177] WIKIPEDIA. E-textiles. https://en.WIKIPEDIA.org/wiki/E-textiles. Last visited on 28/10/2015.
[178] WIKIPEDIA. Speech Synthesis. https://en.WIKIPEDIA.org/wiki/Speech_synthesis. Last visited on 28/10/2015.
[179] Ivona Text to Speech Home Page. https://www.ivona.com/. Last visited on 28/10/2015.
[180] CerecPro. Cerecvoice Home Page. https://www.cereproc.com/en/products/cloud. Last visited on 28/10/2015.
[181] Cepstral Home Page. http://www.cepstral.com/ . Last visited on 28/10/2015.
[182] OATS. eSpeak. http://www.oatsoft.org/Software/espeak-text-to-speech. Last visited on 28/10/2015.
[183] Voxygen Home Page. https://www.voxygen.fr/ Last Visited on 28/10/2015.
[184] Nuance. Loquendo Home Page. http://www.nuance.com/support/loquendo/index.htm Last visited on 28/10/2015.
[185] Nuance. Vocalizer Home Page. http://www.nuance.com/for-business/text-to-speech/vocalizer/index.htm. Last visited on 28/10/2015
[186] Neospeech Home Page. http://neospeech.com/ Last visited on 28/10/2015.

[187] Google Test to Speech. https://play.google.com/store/apps/details?id=com.google.android.tts Last visited on 28/10/2015.

[188] Apple. iOS VoiceOver. http://www.apple.com/accessibility/ios/voiceover/ Last visited on 28/10/2015.

[189] Windows Narrator. http://windows.microsoft.com/en-us/windows-10/getstarted-hear-text-read-aloud. Last visited on 28/10/2015.

[190] GARTNER. Prentice, S., & Ghubril, A. C. .Hype Cycle for Human-Computer Interaction, 2013.

[191] LAU, Hiu-fai Lau. The Future of Virtual Environments: The Development of Virtual Technology. Computer Science and Information Technology 1(1): 41-50, 2013

[192] 10 Top Ten Reviews. Voice Recognition Software Reviews. http://voice-recognition-software-review.toptenreviews.com/ Last visited on 29/10/2015.

[193] GSA U.S .Government wide Section 508 Accessibility Program. http://www.section508.gov/ Last visite don 29/10/2015.

[194] Speaktoit Home Page. Api.ai. http://api.ai Visited on 29/10/2015.

[195] Textshark Home Page. http://www.textshark.com/ Visited on 29/10/2015

[196] Sensory Home Page. Trullyhandsfree. http://www.sensory.com/products/technologies/trulyhandsfree/. Last visited on 29/10/2015.

[197] Sensory Home Page. TrullyNatural. http://www.sensory.com/products/technologies/trulynatural/. Last visited on 29/10/2015.

[198] iSpeech Home Page. http://www.ispeech.org/#/home. Last visited on 29/10/2015.

[199] Digital Syphon Home Page. Sonic Cloud Online Speech. http://www.digitalsyphon.com/index.asp. Last visited on 29/10/2015.

[200] MeMeMe Mobile Home Page. http://www.mememememobile.com/. Last visited on 29/10/2015.

[201] Vocapia Research Home Page. Vocapia. http://www.vocapia.com/. Last visited on 29/10/2015.

[202] WIKIPEDIA. Haptic Interaction. https://en.wikipedia.org/wiki/Haptic_technology Last visited on 29/10/2015.

[203] HARRIS, W. How haptic technology works. http://electronics.howstuffworks.com/everyday-tech/haptic-technology.htm Last visited on 29/10/2015.

[204] COWIE, Roddy, et al. Emotion recognition in human-computer interaction. Signal Processing Magazine, IEEE, 2001, vol. 18, no 1, p. 32-80.

[205] ZENG, Zhihong, et al. A survey of affect recognition methods: Audio, visual, and spontaneous expressions. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2009, vol. 31, no 1, p. 39-58.

[206] CANALES, Lea; MARTÍNEZ-BARCO, Patricio. Emotion Detection from text: A Survey. Processing in the 5th Information Systems Research Working Days (JISIC 2014), p. 37.

[207] DE SILVA, Liyanage C.; MIYASATO, Tsutomu; NAKATSU, Ryohei. Facial emotion recognition using multi-modal information. En Information, Communications and Signal Processing, 1997. ICICS., Proceedings of 1997 International Conference on. IEEE, 1997. p. 397-401.

[208] EL AYADI, Moataz; KAMEL, Mohamed S.; KARRAY, Fakhri. Survey on speech emotion recognition: Features, classification schemes, and databases. Pattern Recognition, 2011, vol. 44, no 3, p. 572-587.
[209] XANTHOPOULOS, Spyros; XINOGALOS, Stelios. A comparative analysis of cross-platform development approaches for mobile applications. En Proceedings of the 6th Balkan Conference in Informatics. ACM, 2013. p. 213-220.
[210] DALMASSO, Isabelle, et al. Survey, comparison and evaluation of cross platform mobile application development tools. En Wireless Communications and Mobile Computing Conference (IWCMC), 2013 9th International. IEEE, 2013. p. 323-328.
[211] EL-KASSAS, Wafaa S., et al. Taxonomy of Cross-Platform Mobile Applications Development Approaches. Ain Shams Engineering Journal, 2015.
[212] HEITKÖTTER, Henning; HANSCHKE, Sebastian; MAJCHRZAK, Tim A. Evaluating cross-platform development approaches for mobile applications. En Web information systems and technologies. Springer Berlin Heidelberg, 2013. p. 120-138.
[213] OTTKA, Sanna, et al. Comparison of mobile application development tools for multi-platform industrial applications. 2015.
[214] Research2Guidance. Cross Platform Tool Becnhmarking 2014.
[215] MARSHALL, R. Magic Quadrant for Mobile Application Development Platforms. Gartner, 23/07/0215.
[216] ZEIMPEKIS, Vasileios; GIAGLIS, George M.; LEKAKOS, George. A taxonomy of indoor and outdoor positioning techniques for mobile location services. ACM SIGecom Exchanges, 2002, vol. 3, no 4, p. 19-27.
[217] MORENO, Daniel; OCHOA, Sergio F. Survey on Resource Positioning. Computer Science Department, University of Chile.
[218] FANTANA, N. K., et al. IoT applications–value creation for industry. Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems, 2013, p. 153-206.
[219] MEIER, H, "The Human Role in Cyber-Physical Systems", European Academy for Industrial Management (AIM), Advances in Industrial cyber-physical systems, 2014, Web: http://www.europe-aim.eu/wp-content/uploads/2012/07/MeierThe-Human-Role-In-Cyber-Physical-Systems.pdf. Last visited on 05/11/2015.
[220] BMSvision bvba. http://www.visionbms.com/ . Last visited on 05/11/2015.
[221] VERMESAN, Ovidiu; FRIESS, Peter (ed.). Internet of things: converging technologies for smart environments and integrated ecosystems. River Publishers, 2013.
[222] JAVAVARDHANA, G. Internet of Things (IoT): A vision, architectural elements, and future directions. Future Generation Computer Systems, 2013, vol. 29, no 7, p. 1645-1660.
[223] ROGERS, Y. Moving on from Weiser's vision of calm computing: engaging ubicomp experiences, in: UbiComp 2006: Ubiquitous Computing, 2006.
[224] CACERES, R. Ubicomp systems at 20: progress, opportunities, and challenges, IEEE Pervasive Computing 11 (2012) 14–21.
[225] GRAIMANN, Bernhard; ALLISON, Brendan; PFURTSCHELLER, Gert. Brain–computer interfaces: A gentle introduction. En Brain-Computer Interfaces. Springer Berlin Heidelberg, 2010. p. 1-27.
[226] ABDULKADER, Sarah N.; ATIA, Ayman; MOSTAFA, Mostafa-Sami M. Brain computer interfacing: Applications and challenges. Egyptian Informatics Journal, 2015, vol. 16, no 2, p. 213-230.

[227] ZOSS, A. Introduction to Data Visualization: About Data Visualization. http://guides.library.duke.edu/c.php?g=289678&p=1930713. Last visited on 09/11/2015.
[228] Juice Labs. Chart Chooser. http://labs.juiceanalytics.com/chartchooser/index.html Last visited on 09/11/2015.
[229] The Extreme Presentation Method. Choosing a Good Chart. http://extremepresentation.typepad.com/blog/2006/09/choosing_a_good.html Last visited on 09/11/2015.
[230] PARKINSON, M. Updated Graphic Cheat Sheet. http://billiondollargraphics.blogspot.com.br/2012/07/updated-graphic-cheat-sheet.html, 22/07/2012, Last visited on 09/11/2015.
[231] VISUALIZING DATA. http://www.visualisingdata.com/resources/ Last visited on 09/11/2015.
[232] DATAVISUALIZATION.CH. Curated List of Visualization Tools. http://selection.datavisualization.ch/ Last visited on 09/11/2015.
[233] FACTS4WORKERS HCI Technologies for Industrial Shopfloor Taxonomy. https://service.projectplace.com/pp/pp.cgi/r1157501527
[234] AGREE, P. E. . Computation and Human Experience, Cambridge University Press, 1997.
[235] BRANDL, P., Lacueva Perez, F.J., Smart Glasses Comparison. FACTS4WORKERS. http://facts4workers.eu/smart-glasses-comparison/
[236] Cicret Bracelet. http://cicret.com/wordpress/. Last visited on 23/11/2015.
[237] PROGlove. http://www.proglove.de/. Last visited on 23/11/2015.
[238] The Rufus Cuff. http://rufuslabs.com/industrial/#. Last visited on 23/11/2015.
[239] LACUEVA PEREZ, F.J., BRANDL, P., Taxonomy of HCI Technologies on a Smart Factory Shop-Floor. http://facts4workers.eu/taxonomyofhcitechnologies/
[240] Handheldgroup. So what is rugged?. https://www.handheldgroup.com/why-rugged-handheld-computers/what-is-rugged/. Last visited on 23/11/2015.

A.Human–Computer Interaction

Human–computer interaction (HCI) is an area of research and practice that emerged in the early 1980s, initially as a speciality area in computer science embracing cognitive science and human factors engineering. The Association for Computing Machinery defines HCI as "*a discipline concerned with **the design, evaluation and implementation** of interactive computing systems **for human use** and with the **study of major phenomena surrounding them***" [5]. One important aspect of HCI is the securing of user satisfaction (or simply end user computing satisfaction). "*Because human–computer interaction studies a human and a machine in communication, it draws from **supporting knowledge** on both the machine and the human side. On the machine side, techniques in computer graphics, OSs, programming languages, and development environments are relevant. On the human side, communication theory, graphic and industrial design disciplines, linguistics, social sciences, cognitive psychology, social psychology, and human factors such as computer user satisfaction are relevant. And, of course, engineering and design methods are relevant.*" For Mountuschi [6], HCI is a multi-disciplinary research area focused on interaction modalities between humans and computers.

Alternatives names for HCI are **computer–human interaction (CHI), man–machine interaction (MMI) and human–machine interaction or interfacing (HMI,** which is sometimes **used to refer to the user interface in a manufacturing or process-control system)**. HCI was automatically represented with the emergence of the computer, or more generally the machine, itself. The reason is clear: Most sophisticated machines are worthless unless men can use them properly. This basic argument simply presents the main terms that should be considered in the **design of HCI: functionality and usability** [7].

Another important concept, namely **user experience (UX),** became associated with usability [6]. UX focuses mainly on parameters related to the user: **satisfaction**, enjoyability, emotional fulfilment, aesthetic appeal and so on. Web interface entails research where the concept of UX has been extended and better defined [10]. Web designers often leverage the UX Honeycomb [9] to identify priorities in the design phase. The honeycomb's seven hexagons represent parameters that must be carefully balanced to provide users with a satisfactory **quality of experience (QoE)** by ensuring that an interface is useful, usable, desirable, findable, accessible, credible and valuable (See Figure 23).

Understanding people’s mental models is another important issue in HCI. Users learn and keep knowledge and skills in different ways that are often influenced by their age and cultural and social backgrounds. Thus, HCI studies aim to bridge gaps between users and new technologies. Efficient, effective and natural forms of HCI can reduce the skills levels needed to use complex devices, which can potentially reduce inequality among people by helping to address an issue in the “digital divide”, which is the gap between those who have access to

ICT technologies and skills to make use of those technologies and those who have neither the access nor the skills.

The term "paradigm", as a way to describe waves of research in a field, originates with Thomas Kuhn's theory of the structure of scientific revolutions [3]. Kuhn describes a model of knowledge based on successive and overlapping waves in which ideas are fundamentally reframed. Following along these lines, a scientific paradigm in HCI would contain: a common understanding of the salient properties of interaction; the types of questions that appear to be both interesting and answerable about those properties of interaction; a set of broad procedures that can be used to provide warrantable answers to those questions; and a common understanding of how to interpret the results of these procedures. These four elements are interdependent and grounded in a deeper common conceptualisation embodied in the examples that are used in schools to teach the field. By contrast, for Kuhn, a paradigm shift is accompanied by a shift in the examples that are considered to be central to the field.

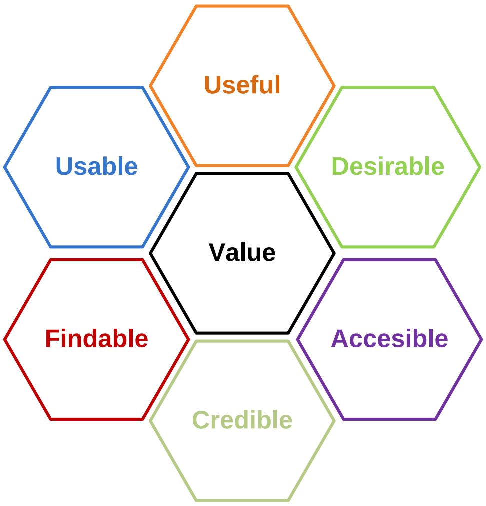


Figure 23: User experience honeycomb

Alternatively, Philip Agre's theory of generative metaphors in technical work suggests that HCI paradigm shifts are pinpointed by **tracing shifts in the underlying metaphor**. For this approach, the **core of each HCI paradigm is a different metaphor of interaction,** which points to the questions that are interesting to ask and the methods to answer them. A paradigm shift occurs when a new generative metaphor is driving new choices of what to research and how, and it can be identified when problems and issues that used to be marginalised move to the centre.

Applying these theories, it is possible to identify three HCI paradigms: **the human factors (man–machine) paradigm, the cognitive science paradigm and the phenomenological matrix paradigm.** These paradigms are briefly introduced in the following paragraphs.

The human factors (man–machine) paradigm is a combination of engineering and human factors. It considers **interaction to be a form of man–machine coupling in ways inspired by industrial engineering and ergonomics, and its goal is to optimise the fit between humans and machines**. Consequently, the questions to be answered focus on identifying problems in coupling and developing pragmatic solutions to them.

The second paradigm, the cognitive science paradigm, is organised around a central metaphor of mind and computer as symmetric, coupled information processors. As human information processing and computer signal processing are analogous, **the primary computer–human interaction task is enabling communication between the machine and the person**. At the centre is a set of information processing phenomena or issues in computers and users, such as *'How does information get in?'*, *'What are the transformations that it undergoes?'*, *'How does it go out again?'*, *'How can it be communicated efficiently?'* etc. At the fringes are phenomena that are difficult to assimilate with information processing: feelings about interaction; the place of interaction; and aspects of everyday life, such as "*What is fun?*"

The movement of the centre of attention towards issues that cannot be covered by the information-processing metaphor required a new paradigm (Harrison [2]). Firstly, **ubiquitous computing** (see Annex A.a.xii: Ubiquitous/Pervasive Computing) **suggests a renewed centrality for the use context of computing**. Secondly, **workplace studies** focus on the **social situation of interaction,** in particular, **the centrality of social, situated actions in explaining the meaning of interaction**. Thirdly, the situation of learning environments and the politics of their evaluation require new metrics for evaluating user satisfaction and even performance, which is not completely covered by K-12 learning goals. Fourthly, non-task-oriented computing makes it difficult to measure efficiency, as it is defined by the first and second paradigms, which require problems to be formalised and expressed in terms of tasks, goals and efficiency. Lastly, yet another set of issues arises from the marginalisation of emotion in classical cognitive work.

Harrison [2] introduced what he called the "***phenomenological matrix paradigm***", the third HCI paradigm to solve the issues mentioned in the paragraph above. It is focused **on the embodiment interaction: The way in which we come to understand the world, ourselves and interaction crucially derives from our location in a physical and social world as embodied actors**. Thinking is not just cognitive, abstract and information-based: **Thinking is also achieved by doing things in the world**. Moreover, it refocuses attention from the single-user/single-computer paradigm on **collaboration and communication through physically shared objects, and it also highlights the importance of risk as a positive aspect of embodied practice** (there is no undo button in the real world). Finally, it reminds us that real-world practice is complex and rich, and it interweaves physical activity and awareness with abstract thoughts, rituals and social interaction in ways that defy a purely informational approach.

The core of the third paradigm **is a phenomenological viewpoint of embodiment in which all action, interaction and knowledge is seen as embodied in situated human actors**. It **considers meaning and meaning construction to be the central focus:** It is constructed on the fly, often collaboratively, by people in specific contexts and situations, and therefore the interaction itself is an essential element of meaning construction. Accordingly, **situated knowledge refers to the idea that people's understanding of the world, themselves and, in the case of HCI, interaction are strongly influenced or perhaps even constructed by varying physical and social situations**. The move to embodiment is a shift towards recognising a plurality of perspectives. **Designing interaction moves from attempting to establish one correct understanding and a set of metrics of interaction to studying the local, situated practices of users while taking into account but not adjudicating the varying and perhaps conflicting perspectives of users.**

Table 32: HCI paradigms summary

	Paradigm 1	Paradigm 2	Paradigm 3
Metaphor of Interaction	Interaction as man–machine coupling	Interaction as information communication	Interaction as phenomenologically situated
Central Goal for Interaction	Optimising fit between man and machine	Optimising the accuracy and efficiency of information transfer	Supporting situated action in the world
Typical Questions of Interest	How can we fix specific problems that arise in interaction?	What are the mismatches that come up in communication between computers and people? How can we accurately model what people do? How can we improve the efficiency of computer use?	Which situated activities in the world should we support? How do users appropriate technologies, and how can we support those appropriations? How can we support interaction without constraining it too much by what a computer can do or understand? What are the politics and values at the site of interaction, and how can we influence those being designed?

The result of considering situated embodiment as crucial is the importance of place in computing. McCullough [108] treats Ubicomp from an architectural perspective and analyses the significance of technologies becoming designed to be adaptable to specific locations, times, social situations and surrounding systems. **Putting interfaces in their place is grounded in the recognition that the specifics of particular contexts greatly define the meaning and the nature of an interaction**. One design strategy is to make the computation and the interface embodied. By **designing the externalities of the interface, the device or system does not have to model every contingency**.

Owing to the variety of potentially valid viewpoints, the evaluation of what makes a system a success can no longer be rooted *a priori* in measures said to be universally valid. Instead, **we must question what it means for a system to be 'good' in a particular context**. **Value-based approaches to HCI** (such as participatory design and value-sensitive design) **have come into use to establish new criteria for success – and therefore of decision making – in system design and evaluation [109]**.

The first and second paradigms **acknowledge context primarily as "those non-technological factors that affect the use of the technology".** The third paradigm considers the context as a central component not only to the problem (if there is any) but also to the design and evaluation. The three paradigms are compared with each other in Table 32: HCI paradigms summary. Because of its emphasis on multiple perspectives, it does not espouse a single, correct set of methods or approaches. Instead, there is a variety of approaches that are embedded in a similar epistemological substrate, like a biological matrix. That is why it is known as **the phenomenological matrix**, a multidimensional characterisation of concerns in which relationships and sequences can be defined as the third paradigm.

The third paradigm should have to confront some **challenges** in order to prove its validity. One of the questions to solve is **how to measure success**. In the second paradigm, measures of success focus on measuring the comparative effectiveness and efficiency of information transfer. Self-reported user satisfaction might suffice, but it is seen as a poor substitute for efficiency. For the third paradigm, some criteria, such as delight, are not seen as legitimate criteria at all. Provoking ideas or causing the reader to consider new possibilities is **not considered a sufficient measure of success**. Furthermore, **balancing the concerns of different stakeholders in a clever way or enabling activity that would otherwise simply not be possible are not sufficient measures of success**.

Another challenge that requires further effort is the recognition of innovation. If HCI wants to be consequential, it must explain important questions. However, many questions cannot be addressed within the second paradigm framework. A nice-looking interface cannot be evaluated on its own terms but has to be measured in functional terms. Don Norman legitimated **emotional design** [110] by demonstrating that good-looking interfaces produce more efficient outcomes. Furthermore, questions about the equivalency of designs rather than differences between them cannot be well explored using statistical methods.

Table 33: Epistemological distinctions between the paradigms contrasts the third paradigm's epistemological commitments with those of the two firsts. While the first and second paradigms emphasise the importance of objective knowledge, the third paradigm sees knowledge as arising from situated viewpoints and often sees the dominant focus on objective knowledge as proof that there is not enough consideration of the complexities of multiple perspectives at the scene of action.

The second paradigm arises out of a combination of computer science and laboratory behavioural sciences that emphasises analytic means and uses generalised models, such as statistical analysis, classification and corroboration. However, the third paradigm sees knowledge as arising out of and becoming meaningful in specific situations, that is, "externalities" are often central figures in the understanding of interaction. This demands new approaches to see interaction as stimulating multiple interpretations in concrete, real-world situations. Moreover, the job of the evaluator will change to identify and track those interpretations, often in collaboration with their 'subjects'.

Table 33: Epistemological distinctions between the paradigms

	Paradigm 1	Paradigm 2	Paradigm 3
Appropriate Disciplines of Interaction	Engineering, programming, ergonomics	Laboratory and theoretical behavioural science	Ethnography, action research, practice-based research, interaction analysis
Kinds of Methods Strived for	Cool hacks	Verified design and evaluation methods that can be applied regardless of context	A palette of situated design and evaluation strategies
Legitimate Kinds of Knowledge	Pragmatic objective details	Objective statements with general applicability	Thick description, stakeholder "care abouts"
How you Know Something is True	You tried it, and it worked	You refute the idea that the difference between experimental conditions is owed to chance	You argue about the relationship between your data and what you seek to understand
Values	Reduce errors Ad hoc is acceptable Cool hacks desired	Optimisation Generalisability wherever possible Principled evaluation is *a priori* better than ad hoc, since design can be structured to reflect the paradigm Structured design better than unstructured Reduction of ambiguity Top-down view of Knowledge	Construction of meaning is intrinsic to interaction activity What goes on around systems is more interesting than what is happening at the interface "Zensign" – what you do not build is as important as what you do build Goal is to grapple with the full complexity around the system

B. Interaction Paradigms

An **interaction paradigm** is a model of HCI, including all the aspects of interaction. We can analyse systems by applying the 5W+H questions: What/How, Where/When and Who/Why. By asking these questions, we found several different kinds of interaction paradigms that, are introduced in following paragraphs according to their relevance for the FACTSW4WORKERS goals.

i. Large-Scale/Mainframe Computing

In the 1950s, programmers were limited in the form of batch files: Complete jobs were submitted on punch cards, and operators were supposed to run it on complex computer hardware [102]. Multi-programmed, batched systems were introduced effectively to utilise various system resources that subsequently extended into time-sharing systems in which the CPU executes multiple jobs. In time-sharing systems, the programmer could interact with the computer in a more reactive and spontaneous manner within the limits of command language, increasing the information processing throughput and programmer's productivity.

ii. Personal Computing/WIMP

Personal computing systems are considerably smaller, less expense and more suitable for office environments than mainframe systems. They train interaction on addressing the single user engaged in a dialogue with the computer in order to carry out a series of tasks.

Initially they adopted the line command metaphor of interaction, but the main reason for its popularisation was the creation of WIMP ("windows, icons, menus and pointer") interfaces. WIMP systems derive from graphical user interfaces. However, while all WIMP systems use graphics as a key element (the icon and pointer elements), and are therefore GUIs, the reverse is not true.

In a WIMP system, a window runs a self-contained programme, isolated from other programmes that (if in a multi-program OS) run at the same time in other windows. An icon acts as a shortcut to an action the computer performs. A menu is a text or icon-based selection system that selects and executes programmes or tasks. The pointer is an onscreen symbol that represents movement of a physical device that the user controls to select icons, data elements etc.

This system style improves HCI by emulating real-world interactions and providing better ease of use for non-technical people. Users can transfer skills at a standardised interface from one application to another.

iii. (Virtual) Network Computing

The concept of **network computing** arose when networks appeared [101]. Network computing aims to give users access to centralised resources from simple and inexpensive devices. These devices act as clients to more powerful server machines (client-server paradigm). Server machines provide applications, data and storage for a user's preferences and personal customisation, while clients are in charge of HCI and minor logic implementations.

As servers' computing capabilities increased, a new concept appeared: **virtual network computing (VNC)**. Server machines supply not only applications and data but also an entire desktop environment that can be accessed from any Internet-connected machine using simple software: a thin client. Whenever and wherever a VNC desktop is accessed, its state and configuration (right down to the position of the cursor) are exactly the same as when it was last accessed.

Both network computing approaches can be considered a natural evolution of WIMP interfaces.

iv. Mobile Computing

Mobile computing is defined [94] as the use of transportable computing devices with mobile communication technologies. Mobile computing is a technology that allows for the transmission of data, voice and video via a computer or any other wireless-enabled device without it having to be connected to a fixed physical link.

Communication issues include ad hoc and infrastructure networks, as well as communication properties, protocols, data formats and concrete technologies. Hardware includes mobile devices or device components. Mobile software deals with the characteristics and requirements of mobile applications.

The Information Systems Audit and Control Association's (ISACA) white paper from 2010 on securing mobile devices defines the following seven types of items as mobile computing devices: smart phones; laptops (portable computers); tablet computers; portable digital assistants (PDAs); portable USB storage devices (portable universal serial bus storage devices); radio and mobile frequency identification devices (RFIDs); and infrared-enabled devices (IrDAs).

The main constraints or limitations of mobile computing are [95][96]: **Mobile elements are resource-poor relative to static elements, and mobility is inherently hazardous.** In our project context, it is of significant importance that people using mobile devices while performing daily tasks are often distracted from them, and therefore the assumption is that they are more likely to be involved in accidents than those who do not use such devices during work. Furthermore, **mobile connectivity is highly variable** in performance and reliability, and **mobile elements rely on a finite energy source**.

v. Wearable Computing

Wearable computers, also known as body-borne computers or wearables, are miniature electronic devices that are worn by the bearer under, with or on top of clothing. This class of wearable technology has been developed for general or special purpose information technologies and media development.

One of the main features of a wearable computer is consistency. There is constant interaction between the computer and the user, i.e., there is no need to turn the device on or off. Another feature is the ability to multi-task. It is not necessary to stop what you are doing to use the device; it is integrated with all other actions. These devices can be incorporated by the user to act like a prosthetic. It can therefore be an extension of the user's mind and/or body.

Wearables have many issues in common with mobile computing, ambient intelligence and ubiquitous computing research communities, including power management and heat dissipation, software architectures and wireless and personal area networks.

vi. Collaborative Computing

Collaborative computing [97], also known as **computer-supported cooperative work (CSCW)**, encompasses the use of computers to support coordination and cooperation of two or more people who attempt to perform a task or solve a problem together [111]. Not surprisingly, systems that have been honed to support group work are referred to as **groupware**. The essence of groupware is the creation of shared workspaces among collaborators.

Collaborative computing sits at the crossroads of many disciplines: multimedia, distributed systems, networking and human factors, to name a few. It is usually conceptualised by considering the context of a CSCW system's use. One conceptualisation is the CSCW matrix [112], which considers work contexts along two dimensions (see Figure 24: CSCW matrix [99]): firstly, whether collaboration is co-located or geographically distributed, and secondly, whether individuals collaborate synchronously (at the same time) or asynchronously (independent of others).

Each quadrant of Figure 24 shows different interactions and possible applications.

vii. Virtual Reality

Virtual reality (VR), which can be referred to as immersive multimedia or computer-simulated life, replicates an environment that simulates physical presence in places in the real world or imagined worlds and lets the user interact in that world. Virtual reality artificially creates sensory experiences that can include sight, hearing, touch, smell and taste.

Most up-to-date virtual reality environments are displayed either on a computer screen or with special stereoscopic displays, and some simulations include additional sensory information and focus on real sound through speakers or headphones targeted towards VR users. Some advanced, haptic systems include tactile information, generally known as force feedback in medical, gaming and military applications. Furthermore, virtual reality covers remote communication environments that **provide virtual presence of users with the concepts of telepresence and telexistence or a virtual artefact (VA),** either through the use of standard input devices such as a keyboard and mouse, or through multimodal devices such as a wired glove or omnidirectional treadmills. The simulated environment can be similar to the real world in order to create a lifelike experience – for example, in simulations for pilot or combat training – or it can differ significantly from reality, such as in VR games.

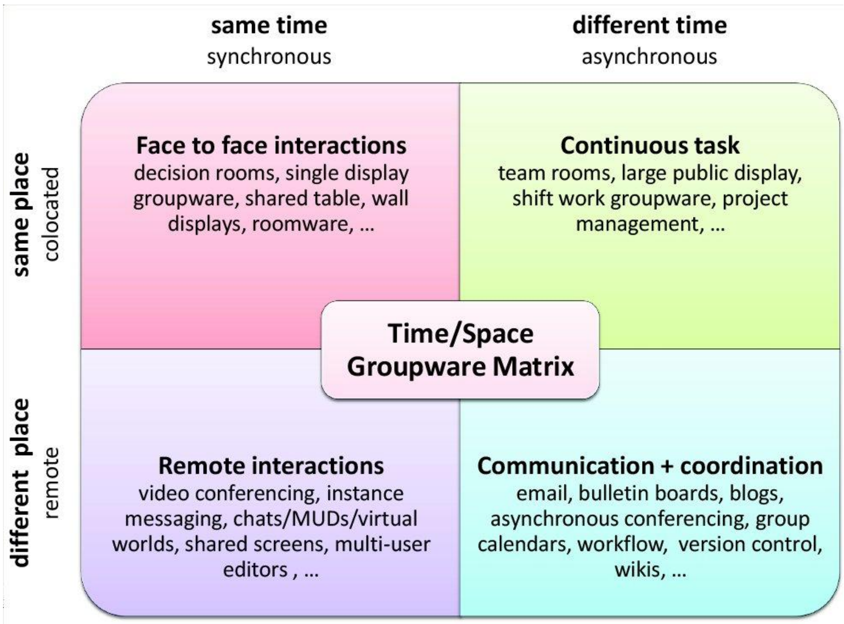


Figure 24: CSCW matrix [99]

viii. Augmented Reality

Azuma [3] provides a commonly accepted definition of AR as a technology that (1) combines real and virtual imagery, (2) is interactive in real time and (3) registers the virtual imagery with the real world. It is this "real world" element that differentiates AR from virtual reality. AR integrates and adds value to the user's interaction with the real world, as opposed to a simulation. AR involves adding information in context to existing reality, such as statistics about a tourist attraction or how a couch will look in someone's living room. Virtual reality pulls the user out of his or her context and replaces the real world with the world of a game or training video [66].

ix. Natural Interaction

Natural interaction is based on a **natural user interface (NUI)**. This is a system for HCI that the user operates through intuitive actions related to natural, everyday human behaviour. A NUI may be operated in a number of different ways, depending on the purpose and user requirements. Some NUIs rely on intermediary devices for interaction, but more advanced NUIs are either invisible to the user or so unobtrusive that they quickly seem invisible.

Some examples and applications of natural user interfaces are touchscreen interfaces (see Chapter 4.2) gesture recognition systems (see Chapter 4.5), speech recognition (see Chapter 4.7.1), gaze tracking (see Chapter 4.6) and brain–machine interfaces (see Chapter 4.10).

x. Multimodal Interaction

Multimodal interaction [105] refers to interaction with the virtual and physical environment through natural modes of communication. It involves all five human senses, and it enables more free and natural communication by allowing users to interface with automated systems in both input and output. Multimodal systems offer a flexible, efficient and usable environment that makes it possible for users to interact through input modalities (i.e., speech, handwriting, hand gesture or gaze), and to receive information by the system through output modalities (speech synthesis, smart graphics etc.).

The combination of different input channels, according to temporal and contextual constraints, in order to allow their interpretation is known as **multimodal fusion**. This process can produce more than one interpretation for each implied modality, and consequently it can produce multimodal ambiguity due to imprecision, noises or other similar factors. Once the ambiguity is resolved, the system returns to the user outputs through various (disaggregated) modal channels arranged according to a consistent feedback as a result of a process called **fission**.

Using the cloud in order to engage shared computational resources and manage the complexity of multimodal interaction represents an opportunity. In fact, cloud computing allows delivering shared scalable, configurable computing resources that can be dynamically and automatically provisioned and released [114].

Two major groups of multimodal interfaces have merged with each other. The first group of interfaces combined various user input modes beyond the traditional keyboard and mouse input/output, such as speech, pen, touch, manual gestures, gaze and head and body movements [115][116]. The most common such interface combines a visual modality with a voice modality; however, other modalities, such as pen-based input or haptic input/output, may also be used. The advantage of multiple input modalities is increased usability. They have implications for accessibility, in particular for people who are [117] "**situationally impaired**" (e.g., wearing gloves) and will simply use the appropriate modalities as desired.

The second group of multimodal systems presents users with multimedia displays and multimodal output, primarily in the form of visual and auditory cues. Proposed benefits of multimodal output system include synergy and redundancy. The information that is presented via several modalities is merged and refers to various aspects of the same process. The use of several modalities for processing exactly the same information provides an increased bandwidth of information transfer [118].

Invisible interface spaces became available through the use of sensor technology. Infrared, ultrasound and cameras are all now commonly used [119]. Transparency of interfacing with content is enhanced, and providing an immediate and direct link via meaningful mapping is possible; thus the user has direct and immediate feedback to input, and content response becomes interface affordance (Gibson 1979).

xi. Adaptive Interfaces

A **user-adaptive [107]** system is interactive and adapts its behaviour to individual users on the basis of processes of user model acquisition and application that involve some form of learning, inference or decision making. This definition distinguishes user-adaptive systems **from adaptable systems,** which the individual user can explicitly tailor to his/her own preferences.

The main functions of an adaptive system are that it: supports system use; takes over parts of routine tasks; adapts the user interface so that it fits better with the user's way of working with the system; helps with system use; mediates interactions with the real world; and controls the dialogue with the user.

Another type of functions an adaptive system also provides are related to supporting information acquisition: helping users to find information; recommending products; tailoring information presentation; and supporting collaboration and learning.

xii. Ubiquitous/Pervasive Computing

Ubiquitous computing is the method of enhancing computer use by making many computers available throughout the physical environment, but making them effectively invisible to the user [103]. The challenge of ubiquitous computing is to create a new kind of relationship between people and computers: one in which the computer would have to take the lead and become vastly better at getting out of the way so that people can just go about their lives. **Ubicomp** attempts to break away from the paradigm of desktop computing to provide computational services to a user whenever and wherever required. Rather than force the user to search out and find the computer's interface, ubiquitous computing suggests that the interface itself can take on the responsibility of locating and serving the user.

A ubiquitous computing system consists of (a) a (possibly heterogeneous) set of computing devices; (b) a set of supported tasks; and (c) some optional infrastructure (e.g., network, GPS location service) on which the devices may rely to carry out the supported tasks.

Unlike traditional desktop applications with a GUI, ubicomp applications are forced to take a rather general view of a system. The emphasis is on combining software components to provide services to the user. **Ubicomp systems are concerned not only with software services but also with devices and how to combine them**.

According to Weiser, ubicomp has two main attributes: interaction with the system is available wherever the user needs it (**ubiquity**); and the system is non-intrusive and integrated into the everyday environment (**transparency**). In Weiser's view, ubiquity denotes the universal availability of computers throughout multiple ubicomp systems in the user's environment.

In ubiquitous computing, the **user mobility** dimension reflects the freedom the user has to move about when interacting with the system. It can be divided into: **constrained mobility,** which allows movement in a well-defined and limited space, and **full mobility,** which defines systems that pose absolutely no constraints on the geographical location of the user.

Transparency (also known as invisibility, embodiment in the environment, intuitiveness, anticipation of the user's intent, affordance and peripheral awareness) applies to the system's interface and **reflects the conscious efforts and attention the system requires of the user, either to operate it or to perceive its output**. **A transparent interface** disappears from the user's focus so it is possible to concentrate on the task at hand.

The transparency dimension can also be divided into two dimensions: **syntactic** and **semantic transparency**. **Syntactic transparency** relieves the user of syntactic tasks; they are introduced by the system itself. When performing a syntactical task, the user does not "do the work" but merely wrestles with the system's specifics (its syntax) in order to perform a "real" semantic task later on. Syntactic transparency also applies to output. When syntactically transparent, a system makes the user aware of its workings in a non-intrusive way. By contrast, **semantic transparency** characterises a system that anticipates the user's intent and performs the task for him/her. Semantic transparency for output happens when the system communicates real-world information (as opposed to information about the system itself) in a way that is not attention-grabbing.

A ubiquitous computing system can be also characterised by the provision of two services: **context awareness; and automated capture, integration and access**. Increased user mobility suggests that applications should adapt themselves based on knowledge of the location. This location can be the position and orientation of a single person, many people, or even of a certain set of devices. Location is a simple example of (physical) **context**, that is, information about people or devices that can be used to modify the way a system provides its services to the user community. Other categories of context include **informational** (what data is the user focused on), **emotional** (how a user feels), **intentional** (what the user wants to do) and **historical** (what the record of context is over time). Context-aware computing aims to provide maximal flexibility for a computational service based on real-time sensing of any of these forms of context.

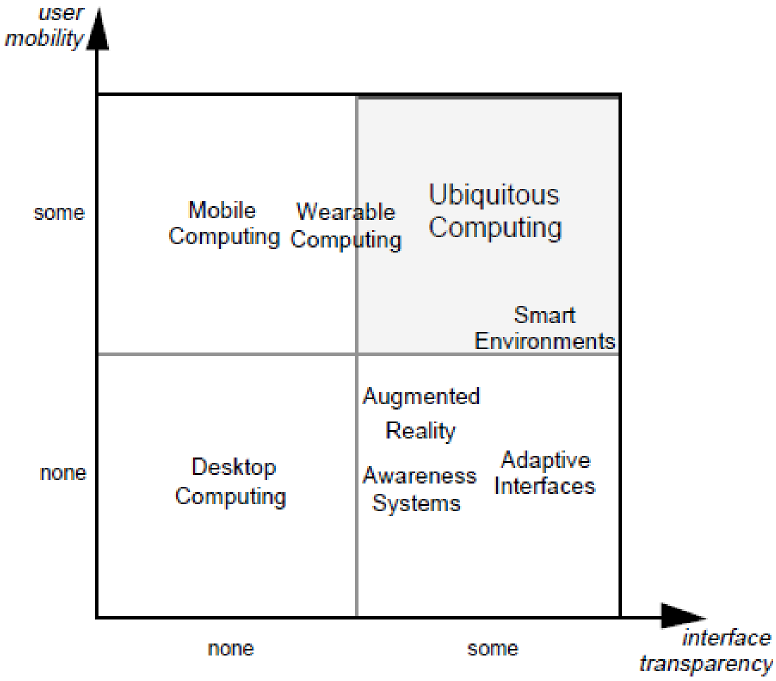


Figure 25: Comparison of ubiquitous computing with other paradigms

The need for context awareness increases as ubicomp applications move towards full mobility. Location awareness is the simplest form of context awareness. Better context awareness is not always best served by higher-precision location services. **Determining the focus of attention of a visitor is most important and can be gleaned from an approximate position and orientation combined with gaze, speech and gesture**. Context awareness is critical to achieving any level of interaction transparency. **Further contextual information, such as the history of interactions by a single user, could then be used to provide semantic transparency that would lead to a precise understanding of which operation the user intends to invoke and automatically call it**.

There is value to using computational resources to augment the inefficiency of human record taking, especially when there are multiple streams of related information. Computational support can also automate explicit and implicit links between related but separately generated streams of information. Finally, a rich record of a group interaction can support access later on to aid in recalling the meaning or significance of past events. Altogether, automated capture, integration and access tools can remove the burden of doing something we are not good at (recording) so that we can focus attention on things we are good at (indicating relationships, summarising and interpreting).

Figure 26 summarises the features and differences between different interaction paradigms. Accordingly, we can follow some of the main issues to be solved and share our insigths with others.For example, privacy is easily the most oft-cited criticism of ubiquitous computing (ubicomp) and may be the greatest barrier to its long-term success. Another issue to be resolved is what 'content' means in a ubiquitous environment. Whereas the interface is clear and distinct in other media environments, ‘content’ takes on a different meaning in a ubiquitous environment.

C. The Model Human Processor

The **model human processor** proposes a simplified view of the human processing system that is involved in interacting with computers (see Figure 26 MHP model). MHP is composed of three subsystems (see Figure 26: MHP model as represented by Baley [19]): perceptual, cognitive and motor. Each subsystem has a processor (characterised by its cycle time) and memory (defined by its capacity and decay time) [20]. Figure 26: MHP model as represented by Baley [19] presents a high-level view of MHP. Although this model does not include the haptic sensory processor and memory, it has easily been extended to include them.

MHP also includes a number of **principles of operation** dictating the behaviour of the system under certain conditions, that is, according to the context. Sometimes systems operate in serial (a key is pressed in response to a stimulus) or in parallel (person driving, talking to passenger, listening to the radio).

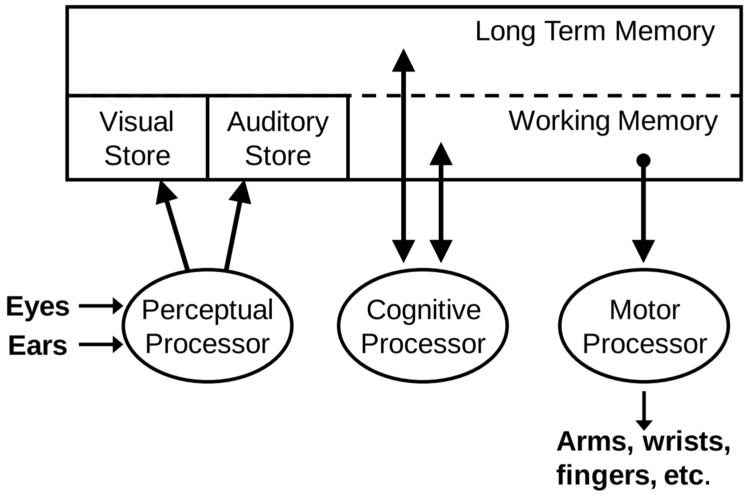


Figure 26: MHP model as represented by Baley [19]

xiii. The Perceptual System

The perceptual system is responsible for transforming the external environment into a form that the cognitive system can process. It is shown by Figure 27 Perceptual system, and it is composed of **perceptual memory** and a **processor**.

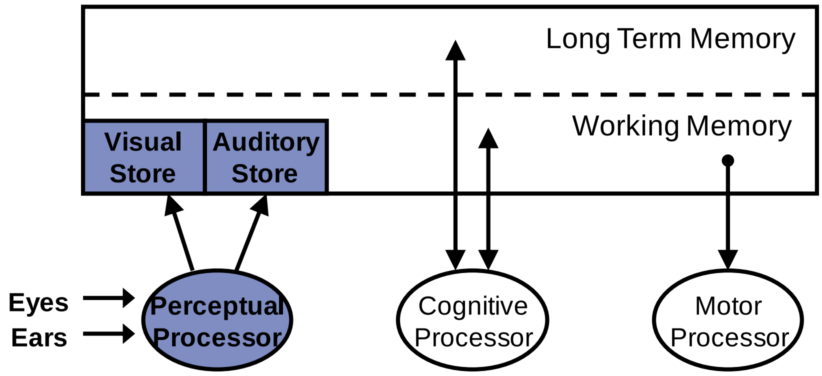


Figure 27: Perceptual system

Shortly after the appearance of the stimulus, representations of stimulus appear in perceptual memory. These sensory memories hold information that is coded physically, that is, as an unidentified, non-symbolic analogue to the external stimulus (i.e., "7" is recognized just as a pattern without any assigned meaning). Shortly after a physical representation of a stimulus appears in one of the perceptual memories, a recognised, symbolic representation of at least part of the perceptual memory contents occurs in the cognitive working memory. If the content of the perceptual memory is complex or has many parts (e.g., an array of letters) and if the stimulus is presented only fleetingly, the perceptual memory trace fades, and the working memory (WM) is filled to capacity before all the items in the cognitive working memory are processed. However, based on physical dimensions, the cognitive processor can specify which portion of the perceptual memory is to be encoded. The decay time is around 200ms for visual storage and 1500ms for audio storage.

The **perceptual processor** codes information in perceptual memory for about 100ms before retrieving the next stimulus. One of the perceptual processor's principles of operation is that the variable processor rate principle determines that the processor cycle time is inversely proportional to the stimulus intensity. Another of the perceptual processor's principles is the **gestalt principles,** which are used to describe how people tend to organise visual elements into groups or unified wholes by applying principles such as similarity, anomaly, continuation, closure, proximity, figure and ground.

Finally, **the encoding specificity principle** [23] provides a framework for understanding how contextual information affects memory and recall. It states that memory is most effective when information available during encoding is also present during retrieval. It explains why a subject is able to recall a target word as part of an unrelated word pair during retrieval much better than if presented with a semantically related word that was not available during encoding. Specific encoding operations determine what is to be stored, which in turn verifies which retrieval cues are effective in providing access to that which was stored. Encoding at the time of perception has an impact on what kind of and how information is stored. The type and order of coding is influenced by gestural principles [22] and the focus of attention that direct the processing or filter information [20].

xiv. Cognitive System

In a very simple view, the cognitive system merely serves to connect inputs from the perceptual system to the right outputs of the motor system. It is shown in Figure 28: Cognitive system. It uses contents of **WM** and **long-term memory (LTM)** to make decisions and schedule actions with the motor system. It is composed of these two memories and the **cognitive processor.**

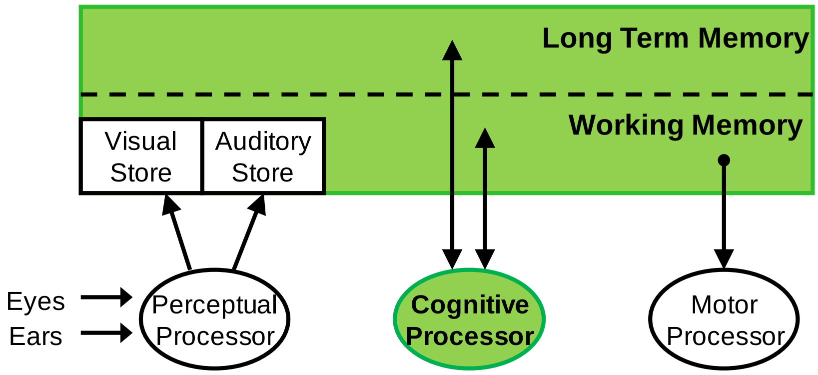


Figure 28: Cognitive system

WM holds the information that is currently under consideration, as well as intermediate products of thinking and representations produced by the perceptual system. Functionally, WM is where all the mental operations obtain their operands and leave their outputs. Structurally, WM consists of activated sections of LTM that are called ***"chunks"*** **[24]**: hierarchical symbol structures that have an individual in common and 7 +/- 2 chunks active at any given time. What constitutes a chunk is the function of the user and of the task, but it depends on the contents of the user's LTM. They can be related to other chunks, and they are activated associatively. A chunk's **decay time** depends on the number of chunks: It is around 7 s for three chunks, but the time is highly variable. The decay time is explained by the **discrimination principle:** A chunk is less accessible because new chunks are activated, but it is also influenced by the number of chunks that are the object of the recall or to the number of candidates that exist in memory relative to retrieval cues.

LTM holds the user's mass of available knowledge. It consists of a network of related, associated chunks, which creates a semantic network. These chunks are accessed associatively by from the contents or WM. Its contents comprise not only facts but also procedures and history. LTM is fast to read but slow to write, because items cannot be added directly. Rather, items in WM have a certain probability of being retrievable later from LTM. The more associations the item has, the greater its probability is of being retrieved. If a user wants to remember something later, his/her best strategy is to attempt ways to avoid interference with other items.

LTM has infinitive storage capacity, but retrievals of a chunk might fail because effective associations cannot be found or (because of the **discrimination principle**) there are similar associations to several chunks interfering with the retrieval of the target chunk.

To be stored in LTM, information from the sensory memories must ultimately be encoded in a symbolic form (the "7" pattern is the digit seven). When information in the WM becomes part of the LTM, the precise way in which it and the coincident WM contents were encoded determines the cues that will be effective in retrieving the item later on. This is known as the **encoding specificity principle**: Specific operations performed on given data determine what is stored, and what is stored determines the retrieval cues that are effective in providing access to what is stored.

The **cognitive processor** is based on the **recognise–act cycle.** Every cycle, which lasts around 70 ms, the contents of WM initiate associatively linked actions in LTM (**"recognise"**). In turn, these actions modify the contents of WM (**"act"**), which sets the stage for the next cycle. Plans, procedures and other forms of extended organised behaviour are built up out of an organised set of recognise–act cycles.

The cognitive system is organised by the **uncertainty principle** and the **variable rate principle**. The **uncertainty principle** establishes that decision time increases along with uncertainty about the judgement to be made, as it requires more cognitive cycles. The **variable rate principle** determines that cycle time is shorter when greater effort is required because of increased task demands or information loads; it also diminishes with practice, following the power law of practice: $T_n = T_1 * n^{-\alpha}$, where α is a learning constant.

xv. Motor System

The motor system, represented in Figure 29: Motor system, translates thoughts into actions by activating patterns of voluntary muscles. Traditionally, computer users' two most important sets of effectors have been the arm–hand–finger system and the head–eye system.

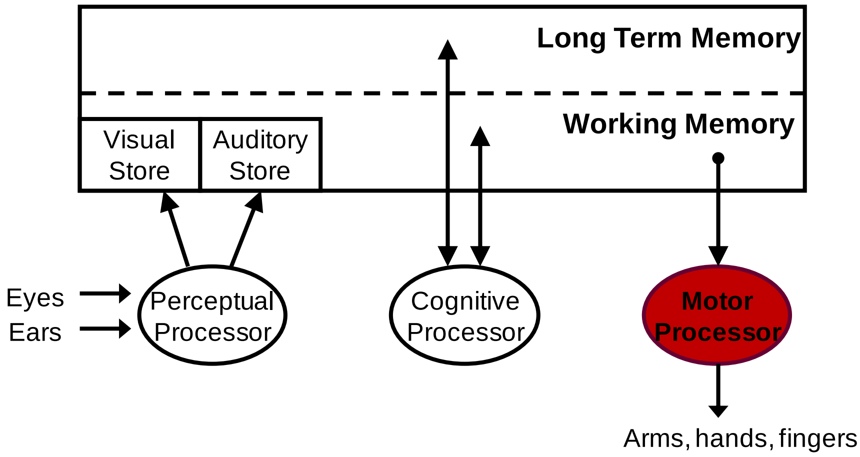


Figure 29: Motor system

Motor Processor is feed by WM. It controls the movements of body. Movement is composed of discrete micro-movements with duration of about 70ms and a Cycle Time of about 70ms. Some of these movements, behavioural acts such as typing and speaking, are chained using special chunks of memory, which are known as caches.

D.Human “*Input–Output* Channels”

In interaction with a computer, human input is the data output by the computer, and vice versa. Input in humans occurs mainly through the senses, and output through the motor controls of the effectors. Vision, hearing and touch are the most important senses in HCI. The fingers, voice, eyes, head and body position are the primary effectors [22].

xvi. Vision

The eye can perceive size and depth using the visual angle: If two objects are at the same distance from the eye, the larger one will have a larger visual angle. Visual acuity is a person’s ability to perceive small details. If the visual angle is too small, the detail will not be perceived. The minimum visual angle is approximately 5 seconds of arc. However, according to the law of size constancy, our perception of size relies on more factors than the visual angle, for example, the perception of depth. Depth can be perceived through various cues, e.g., indications in the visual context about an object’s distance and familiarity with the size of the object. Perception of size and depth are highly intertwined.

The perception of brightness is a subjective reaction to levels of light emitted by an object: Luminance contrast is related to luminance, since it is the function of the luminance of the object and the background. Visual acuity increases with increased luminance. However, on screen, flicker also increases with luminance. The eye perceives colour because the cones are sensitive to the light of different wavelengths. It is important to remember that between 3 and 4% of the fovea is sensitive to blue, which makes blue acuity lower.

The context in which an object appears allows our expectations to clearly disambiguate the interpretation of the object. However, it can also create optical illusions, for example in the Muller–Lyer illusion (see Figure 30).

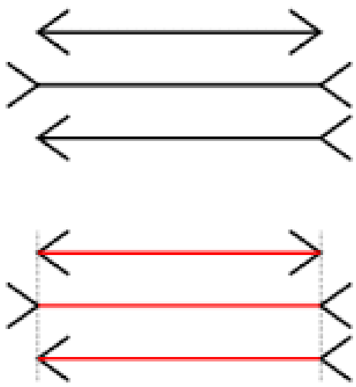


Figure 30: Muller–Lyer illusion

Reading comprises several stages. Firstly, the visual pattern of the word is perceived. Secondly, it is decoded with reference to an internal representation of language. Finally, the word is processed as part of the sentence or phrase using syntactic and semantic analysis. During the first two stages, the eyes make saccades (jerky movements), followed by fixations. The eye moves both forwards and backwards over the text (regressions), and these movements increase when the text is more complex.

xvii. Hearing

The human ear can hear frequencies from 20 Hz to 15 kHz. The sound we perceive is (selectively) filtered, which is illustrated at a cocktail party: We notice when our name is spoken, even in a noisy room.

Sound (vibrations) has a number of characteristics. The pitch is the frequency of the sound. The higher the frequency, the higher the sound. The loudness corresponds to the amplitude of the sound. Timbre relates to the type of the sound, independently of frequency and amplitude.

xviii. Touch

The apparatus of touch (haptic perception) is not localised. Stimuli are received through the skin, which contains various types of sensory receptors. Mechanoreceptors, responding to pressure, are important in HCI. There are two kinds of mechanoreceptors: **rapidly adapting mechanoreceptors**, which respond to immediate pressure and stop responding if continuous pressure is applied, at which point the **slowly adapting mechanoreceptors** start to respond. Some areas of the body have greater sensitivity/acuity than others. This can be measured using the two-point threshold test.

A second aspect of haptic perception is kinesthesis: awareness of the position of the body and limbs thanks to receptors in the joints. There are three types: **rapidly adapting** (respond when moving limbs in a certain direction), **slowly adapting** (respond to movement and static position) and **positional** receptors (only respond to static positions).

xix. Movement

When making movements, a stimulus is received through the sensory receptors and transmitted to the brain. After processing, the brain tells the appropriate muscle to respond. The movement time depends on the physical characteristics of the subjects. The reaction time varies according to the sensory channel through which the stimulus is received.

Fitts’s law [120] [12] studies human movement, and it predicts the required time to rapidly move to a target area as a function of the ratio between the distance to the target and the width of the target. This is formalised as:

$$MT = a + b \log2 (distance=size + 1)$$

where MT is the movement time, a and b are empirical constants.

Fitts’s law is used to model the act of pointing, either by physically touching an object with a hand or a finger, or virtually, by pointing to an object on a computer monitor using a pointing device. It has been applied under a variety of conditions, with many different limbs (hands, feet, the lower lip, head-mounted sights, eye gaze), input devices, physical environments and user populations (young, old, special educational needs and drugged participants).

E. Thinking: Reasoning and Problem Solving

Thinking can require different amounts of knowledge. It can be divided into reasoning and problem solving.

xx. Reasoning

Reasoning is the process in which knowledge is used to draw conclusions or infer something new about the domain of interest. There are different types of reasoning:

- Deduction: Deductive reasoning derives the logically necessary conclusion from the given premises. The human deduction is weak at the points where truth and validity clash.
- Induction: Inductive reasoning is generalising from given cases and inferring information about possible cases. In practice, induction is used to fill in missing details by reasoning.
- Abduction: Abductive reasoning starts with a fact and works towards the action or state that caused it. Abduction is used to derive explanations for the events we observe.

xxi. Problem Solving

Problem solving is the process of finding a solution to an unfamiliar taste by using (adapting) the knowledge we have. There are different views on problem solving:

- Gestalt theory: The gestalt theory states that problem solving is both productive and reproductive; insight is needed to solve a problem.
- Problem space theory: The problem space comprises problem states, and problem solving involves generating these states using legal state transition operators. People use these to move from the initial state to the goal state. Heuristics (e.g., means-end analysis) are employed to select the right operators.
- Use of analogy: Problems are solved by mapping knowledge relating to a similar known domain to the new problem: analogical mapping.

xxii. Skill Acquisition

Experts often have a better encoding of knowledge: Information structures are fine-tuned at a deep level to enable efficient and accurate retrieval. According to the ATC model, these skills are acquired through three levels:

- The learner uses general purpose rules that interpret facts about a problem (slow, memory-intensive).
- The learner develops rules specific to the task by using proceduralisation.
- The rules are tuned to speed up performance by using generalisation.

xxiii. Hick's Law

Hick's law or the Hick–Hyman Law [121] determines the time it takes for a person to make a decision as a result of the possible choices he or she has: Increasing the number of choices will increase the decision time (exponentially). The Hick–Hyman law assesses cognitive information capacity in choice reaction experiments. The amount of time taken to process a certain amount of bits in the Hick–Hyman law is known as the rate of gain of information.

F. 3D Printing

3D printing, also known as additive manufacturing, is one of many various processes used to synthesise a three-dimensional object [142]. In 3D printing, successive layers of material are applied in a process controlled by a computer system. These objects can be of almost any shape or geometry and are produced from a 3D model or other electronic data sources. A 3D printer is a type of industrial robot.

Table 34: 3D Printing technologies and materials

Type	Technologies	Materials
Extrusion	Fused deposition modelling (FDM) or fused filament fabrication (FFF)	Thermoplastics, eutectic metals, edible materials, rubbers, modelling clay, plasticine, metal clay (including precious metal clay)
Lightly Polymerised	Robocasting or direct ink writing (DIW)	Ceramic materials, metal alloy, cermets, metal matrix composite, ceramic matrix composite
	Stereolithography (SLA)	photopolymer
Powder Bed	Digital light processing (DLP)	Digital light processing (DLP)
	Powder bed and inkjet head 3D printing (3DP)	Almost any metal alloy, powdered polymers, plaster
	Electron-beam melting (EBM)	Almost any metal alloy including titanium alloys
	Selective laser melting (SLM)	Titanium alloys, cobalt chrome alloys, stainless steel, aluminium
	Selective heat sintering (SHS)[26]	Thermoplastic powder
	Selective laser sintering (SLS)	Thermoplastics, metal powders, ceramic powders
	Direct metal laser sintering (DMLS)	Almost any metal alloy
Laminated	Laminated object manufacturing (LOM)	Paper, metal foil, plastic film
Wire	Electron-beam freeform fabrication (EBF3)	Almost any metal alloy

Several different 3D printing processes have been invented since the late 1970s, and a large number of additive processes are now available. The main differences between processes concern the way in which layers are deposited to create parts, and the materials that are used. Some methods melt or soften the material to produce the layers, for example selective laser melting (SLM) or direct metal laser sintering (DMLS), and in laminated object manufacturing (LOM), thin layers are cut to shape and joined together (e.g., paper, polymer, metal etc.). Each method has its own advantages and drawbacks, which is why some companies offer a choice of powder and polymer for the material used to build the object. Table 34: 3D Printing technologies and materials summarises 3D printing technologies.

Important features of 3D printers are [144]: build area (the maximum size of an object that the 3D printer can build); extruders (business end of filament printers, where the printing material is melted and extruded to lay down each layer of an object); filament width (consumer 3D printers use a plastic filament, which is available in two widths: 1.75 mm and 3 mm; print speed (which depends on the printing technology, the type of material and the complexity of the model); horizontal, XY or feature resolution (it is the smallest movement that the extruder or print head can make within a printed layer, a smaller horizontal or XY resolution means more fine detail in prints); vertical Z resolution or layer thickness (it is the minimum thickness of a layer that the 3D printer can lay down in one pass).

G. BCI

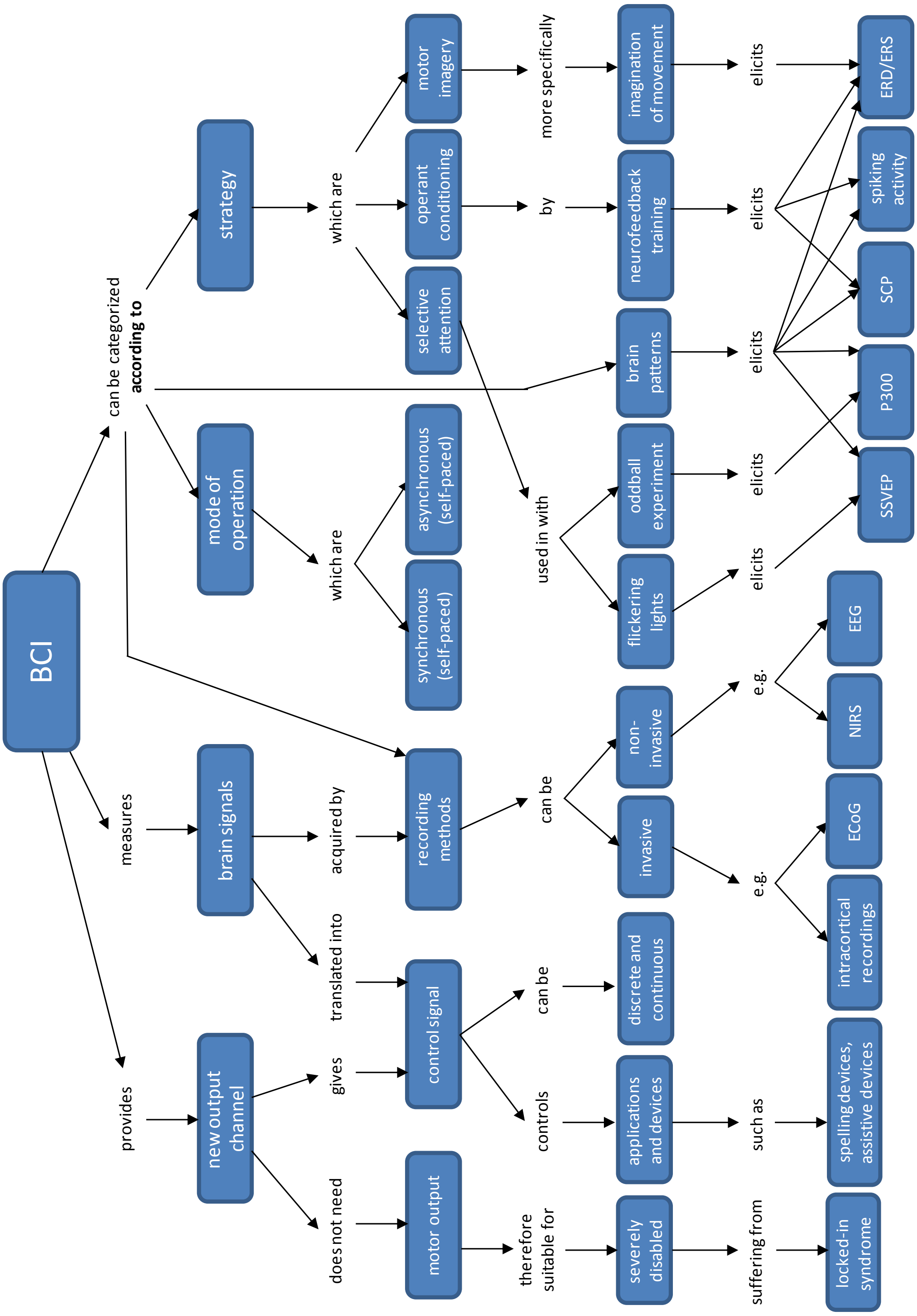


Figure 31: BCI concept map [225]

ABOUT THE PROJECT

The high ambition of the project FACTS4WORKERS is to create Factories of the Future with a pervasive, networked information and communication technology that collects processes and presents large amounts of data. These smart factories will autonomously keep track of inventory, machine parameters, product quality and workforce activities. But at the same time, the worker will play the central role within the future form of production. The ambition of the project is to create »FACTories for WORKERS« (FACTS-4WORKERS), to strengthen human workforce on all levels from shop floor to management since it is the most skilled, flexible, sophisticated and productive asset of any production system and this way ensure a long-term competitiveness of manufacturing industry. Therefore a serious effort will be put into integrating already available IT enablers into a seamless and flexible Smart Factory infrastructure based on work-centric and data-driven technology building blocks.

These solutions will be developed according to the following four industrial challenges which are generalizable to manufacturing in general:

- Personalized augmented operator,
- Worked-centric rich-media knowledge sharing management,
- Self-learning manufacturing workplaces,
- In-situ mobile learning in the production.

PROJECT PARTNERS

The FACTS4WORKERS project is composed of 15 partners from 8 different European countries:

Virtual Vehicle Research Center	Austria
Hidria TC Tehnološki center d.o.o.	Slovenia
Universita degli Studi di Firenze, Department of industrial Engineering	Italy
Technische Universität Wien	Austria
ThyssenKrupp Steel Europe AG	Germany
Hidria Rotomatika d.o.o., Industrija Rotacijskih Sistemov	Slovenia
iMinds VZW	Belgium
Sieva d.o.o.	Slovenia
University of Zurich, Department of Informatics	Switzerland
Thermolympic S.L.	Spain
EMO-Orodjarna d.o.o.	Slovenia
Evolaris Next Level GmbH	Austria
Itainnova - Instituto Technologico de Aragon	Spain
Schaeffler Technologies AG & Co. KG	Germany
Lappeenranta University of Technology	Finland

PROJECT COORDINATOR / CONTACT:

virtual vehicle

VIRTUAL VEHICLE Research Center
Inffeldgasse 21A
8010 Graz, AUSTRIA

Tel.: +43-316-873-9077
Fax: +43-316-873-9002
E-Mail: facts4workers@v2c2.at

FOLLOW US AT:

FACTS4WORKERS
@FACTS4WORKERS
facts4workers-project

Technology Monitoring: Report on Information Needed For Workers in the Smart Factory

D2.1, Technology Monitoring: Report on Information needed for the Industrial Challenges Workers with Taxonomy is part of the work in progress of “FACTorieS for WORKERS” (FACTS4WORKERS) project and more concretely of the task T2.1 of WP2.

D2.1 is the result of the work of T2.1. Its final objective is to create a vision of the current and future developments of HMI technologies and paradigms that will allow other WP2 tasks to obtain the maximum benefit when implementing HCI building blocks, as well as support future technologies adaptation as they become available during the project execution. D2.1 will also provide a general evaluation of existing technologies by considering their applicability on the factories’ shop floor but always observing the project objectives and industrial challenges reflected in the project proposal. The evaluation of the technologies will be provided as a taxonomy of technologies that will be evaluated on a TRL-based scale. The taxonomy will be updated in subsequent versions in order to track the technology maturity evolution during the project’s life. It will also comment on the observed state of technology.