This project has received funding from the European Union's Horizon 2020 research
and innovation programme under grant agreement Nº 101017331

D6.4 Models of emergency prediction and handling v1

Deliverable No.	D6.4	Due Date	28/02/2023
Description	This deliverable includes the details of data processing and management, "smart" execution engine of the automation include recommendations, alerts, automated rules and notifications.		
Type	Report	Dissemination Level	PU
Work Package No.	WP6	Work Package Title	High Level Ecosystem for AI Operations
Version	1.0	Status	Final

Authors

Name and surname	Partner name	e-mail
Pablo Lombillo	MYS	plombillo@mysphera.com
Francesca Manni	PEN	francesca.manni@philips.com
Luís Carrascal	INETUM	luis.carrascal@inetum.com
Antonio Gamito	INETUM	antonio-jesus.gamito@inetum.com
Sofía Granda	INETUM	sofia.granda@inetum.com
Alex Barnadas	INETUM	alex.barnadas@inetum.com

History

Date	Version	Change
01/07/2022	0.1	Initial ToC
20/07/2022	0.2	First version
09/12/2022	0.3	MYS Contribution
10/01/2023	0.4	PEN Contribution
01/02/2023	0.5	Merge and INETUM Contribution
03/02/2023	0.6	Ready for peer review
21/02/2023	0.7	Reviewed by UDGA and MYS
26/02/2023	0.8	Ready for quality check
28/02/2023	1.0	Final version

Key data

Keywords	Disaster preparedness, computer vision, COVID management, IoT
Lead Editor	Alex Barnadas (INETUM). Previously FORTH, as it says in the DoA
Internal Reviewer(s)	Adrian Quesada Rodriguez (UDGA), Pilar SALA (MYS)

Abstract

This deliverable reports the efforts made to develop the task T6.2 (data interpretation and emergency management handling) of WP6 (High level ecosystem for AI operations) regarding the reporting period from month 7 to 24. Risk management is a subject that must be approached by all the organisations but specially the hospitals. This deliverable D6.4, has the responsibility to create a foothold to the systems that the hospitals have regarding the management and handling of emergencies by including innovative and state of the art technologies in the solutions. In this deliverable a first point of view is created so the techniques available to develop this framework is stated. This deliverable counts also on the work done in deliverable D6.1 and D6.2 that perform the data analysis and visualization of the outcome data that the emergency prediction systems provide.

This deliverable counts on a second release (D6.5 Emergency prediction and handling v2) that will improve the solutions and includes new ones that will be developed in collaboration with the pilots.

Statement of originality

This deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

Table of contents

List of tables

List of figures

Acronyms

Acronym	Explanation
FL	Federated Learning
DL	Deep Learning
RTSP	Real-Time Streaming Protocol
NMS	Non-Maximum Suppression
IoU	Intersection over Union
CNN	Convolutional neural network
EM	Emergency Management
GIS	Geography Information Systems
BIM	Building information System
RTLS	Real time location System
GPS	Global Positioning Systems
HVAC	Heating – Ventilation - Air conditioning
BLE	Bluetooth Low Energy

1 Introduction

1.1 Deliverable context

The methodologies applied in the Use Case of disaster preparedness, according to Hendrickx et al.[1] starts from the point of asking the pilot hospitals which disasters or emergencies they are concerned about; from there define which ones might concern the platform. Once defined the problems that might concern ODIN a few solutions are purposed and from then a one or multiple viable solutions will be presented to the pilots.

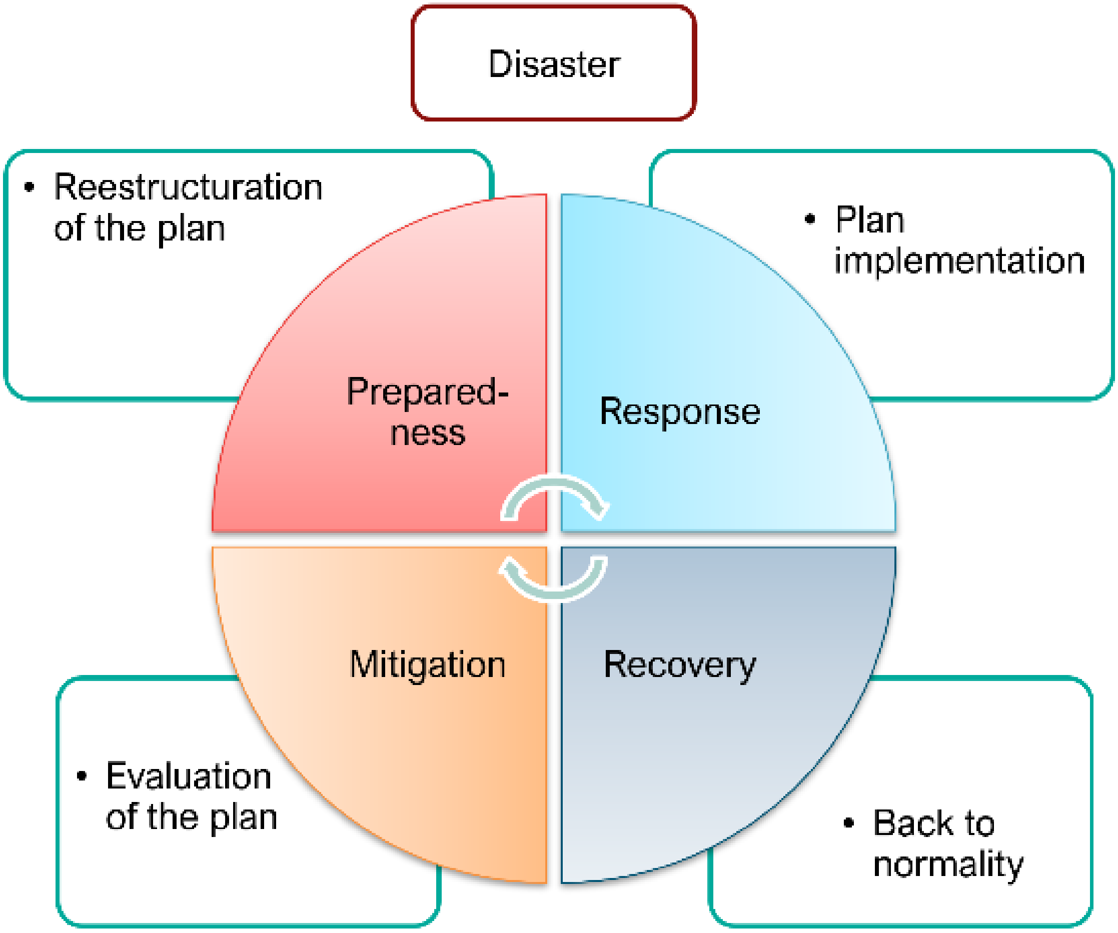

Figure 1: Disaster preparedness flow diagram.

1 Principles of hospital disaster management: an integrated and multidisciplinary approach

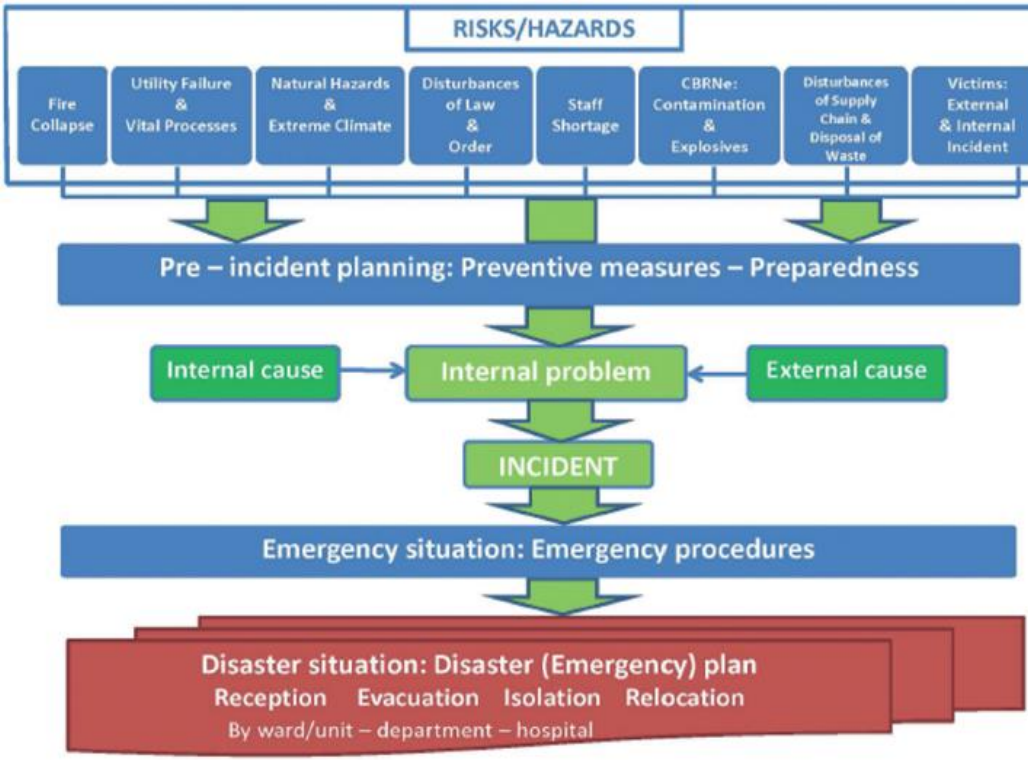

Figure 2: Disaster preparedness pipeline.

As it might seem obvious, this document is about the Preparedness part of the diagram in *Figure 1: Disaster preparedness flow* diagram or the Pre-Incident planning in *Figure 2: Disaster preparedness* pipeline in but the main objective is to make a faster response and an easier recover of the disasters stated to be solved. To do so it is intended to accomplish the rules of the ISO 21110[2] and also to keep an eye on the policies developed by the HERA[3] initiative.

As it is conceived now, the disaster preparedness allows to give an estimated best solution to the problem identified but with the solutions purposed it is intended to create a paradigm shift on how the hospitals approach to disaster preparedness. These solutions will be inspired in state-of-the-art simulations in Emergency Medicine (EM).

An important step in this plan is always to keep the eWorkers aware of the plan since there is no point on making a plan but without a trained staff to execute it and guide everyone to control the situation. For this reason it is suggested that the conclusions of this reference use case are mentioned in the formation of the staff of the hospital regarding deisaster preparedness.

[2] International Organization for Standardization (ISO, 2019). Information and documentation - Emergency preparedness and response (ISO/DIS Standard N0. 21110). Retrieved from https://www.iso.org/standard/69922.html

[3] European Health Emergency Preparedness and Response Authority (HERA). https://ec.europa.eu/info/law/better-regulation/have-your-say/initiatives/12870-European-Health-Emergency-Preparedness-and-Response-Authority-HERA-_en

It might seem difficult to talk about disaster preparedness without mentioning the COVID-19, which has added a new layer of awareness that must be considered in the modern disaster plans, that is why even though the viral transmission measures are every day being less important, the measures will still be taken very seriously in ODIN's Plans.

1.2 Objectives

The main objective of the reference use case C is to prepare hospitals to the considered disasters, but also to increase patient and staff safety and security. For this reason, the following applications have been considered.

Disaster preparedness: COVID measures and sanitary safety

COVID measures, such as minimum interpersonal distance or wearing a mask, are now globally accepted common practices and rules in every hospital sometimes can be hard to make sure everyone is following them. The use of AI could lead to an easier way to remind everyone that these rules are still to be followed by everyone.

Safety: Evacuation risks

Chaotic evacuations is a scenario that can end up being catastrophic, this might be due to the lack of information on the capacity in every single room. To mitigate this risk, we purpose to track the amount of people in each floor/room and trigger an alarm at the moment that its limit is exceeded to keep the capacity of the hospital in its safe values. This way we can have a more precise control on how many people are in the hospital and how many more can enter in a defined area.

Safety: Fall risk

Someone falling might be obvious if there is people around, but it could be the case that people suddenly faint or fall asleep in the middle of the night and might pass a long time before someone notices. In this reference use case several methods are being considered to detect fallen people automatically.

Security: Forgotten objects

Forgotten objects might be a source of danger in a hospital, for example if they are in the middle of an emergency department corridor and can sometimes be treated as suspicious packages as well. With the help of AI forgotten object can be detected very easily and can be returned to its owners before they leave the building.

1.3 ODIN context

Table 1: Deliverable context

PROJECT ITEM IN THE DOA	RELATIONSHIP
Project Objectives	D6.4 is strongly related to ODIN's Objective 1 (O1) that aims "to deliver an open and secure decentralized platform supporting a suite of federated services and Key Enabling Resources (KERs) empowered by robotic solutions, augmented by IoT environments, and enhanced by extensible specialized AI".
Exploitable results	The results of this deliverable will be directly exploited by WP3 (Platform integration, Privacy, Security and Trust + knowledge + cognition) and WP7 (ODIN pilots design, deployment, evaluation and validation).
Workplan	This deliverable is part of the progress of the task T6.2 (Data interpretation and emergency prediction handling) regarding the reporting period of month 7-24 and is the first version of Emergency prediction and handling. The progress of the disaster preparedness is strongly related to the use case that the pilots consider relevant in their dependencies. Thus, the deliverable will be in a kick-off for the pilots to perform AI enhanced disaster preparedness plans.
Milestones	D6.4 is linked to the Milestone MS2 (ODIN technologies catalogue defined and full version of the platform and IoT robotics and AI components).
Deliverables	The current deliverable is related to D6.1 (The data model ecosystem for AI operations and modules implementation) and D6.2 (Data results interpretation and data integration services) that provide the data results analysis and interpretation for the AI-based models. D6.4, is also related to D7.1 (Pilot Studies Use Case Definition and Key Performance) where the pilots' requirements and needs where described in detail. Additionally, D6.4 is related to D3.11 (ODIN Platform v2) that defines the AI as a KER in the ODIN platform and its connection to HIS. And also D3.5 where it is defined the security, privacy and trust.
Risks	The potential risks in this deliverable are mainly Risk 2 (Technical problems during component/module development), Risk 4 (Risk of time consuming due to multiple technology) and Risk 9 (Legal restrictions imposed in the execution of the trials).

2 AI inside ODIN's architecture

The AI algorithms in which is based the emergency handling and prediction systems will have to run on premises due to the unavailability to take the image stream out of the hospital. It is then convenient to set up a way to improve the models used by the system in which all the hospitals benefit at the same time with the information they gather but without sharing the data itself.

2.1 Federated learning platform

Federated learning (FL) is a machine learning-based technology that enables to build AI models across different sites without moving the data, thereby preserving health-data privacy[4, 5].

In a FL setup, multiple decentralized sites train a shared model on private data and send model updates to a coordinating server, which is responsible for the pipeline harmonization. Once all the model updates are received, the coordinator performs the model aggregation and sends back the updated information to each local site. This process continues until a desired level of performance, or a pre-configured number of training rounds, is reached[6,7]. In this way, FL allows research centres, hospitals and organization to overcome existing barriers related to data availability as well as data privacy and security.

Although FL has been already investigated and deployed in several prominent use cases, its adoption is still limited due to a lack of key functionalities needed for providing practical solutions, such as scalability, communication protocols, user configuration, data security, aggregation methodologies and production readiness[5,8]. A recent review by Lo et al. (including 231 papers from 2016-2019) shows that the most studied limitations in FL are the communication, the model performances, the system heterogeneity and scalability[4,6].

Concerning data security in FL, literature studies mostly discuss about the possibility of information leakage from the local model gradient updates[4]. In FL, private information can be extracted by tracing the gradients of a DL model[4]. This data security issue is addressed by employing privacy-preserving methods, such as differential privacy. However, the latter should be used in combination with device secure multiparty computation methods to ensure data privacy at worker side by adding noise to model parameters[9]. In the context of the ODIN project the security and privacy is stated in D3.5 and the FL is a way to be consistent with it.

[4] N.Rieke, J. Hancox, Milletari Li, H. R. Roth, F. Albarqouni, et al. 'The future of digital health with federated learning.' NPJ digital medicine', 3(1), 1-7, 2020.

[5] Q. Li, Z. Wen, Z. Wu, S. Hu, N. Wang, Y. Li, et al. 'A survey on federated learning systems: vision, hype and reality for data privacy and protection. arXiv preprint arXiv, 1907.09693, 2019.

[6] S. Kit LO, Q. LU, C. WANG et al. 'A systematic literature review on federated machine learning: From a software engineering perspective.' ACM Computing Surveys (CSUR), vol. 54, no 5, pp. 1-39, 2021.

[7] Rachakonda, A. S., Moorthy, B. S., Jain, C. A., Bukharev, D. A., Bucur, E. A., Manni, F. F., ... & Mendez, I. N. I. (2022). Privacy enhancing and scalable federated learning to accelerate AI implementation in cross-silo and IoMT environments. IEEE Journal of Biomedical and Health Informatics.

[8] F. Sattler, S. Wiedemann, K. Müller, W. Samek, 'Robust and Communication-Efficient Federated Learning', Non-i.i.d. Data. IEEE Transactions on Neural Networks and Learning Systems.

[9] T. Ryffel, A.Trask, M. Dahl, B. Wagner, J. Mancuso, D. Rueckert, J. P. Palmbach, 'A generic framework for privacy preserving deep learning'. arXiv:1811.04017 [cs.LG], 2018.

Model performance is also largely investigated in FL, by evaluating different approaches and solutions. The dependency on the number of participants and data size of each participant in the training process are the most investigated attributes[4].

Communication costs are improved in the FL process by sharing model updates instead of raw data. However, the number of workers, the data size, the worker drop out should be considered to deploy FL in both cross-devices and cross-silos as they can affect the communication. These aspects are currently studied to investigate their impact on the model performances and the overall training time[10].

Lastly, FL addresses heterogeneity by aggregating local models trained at each site. However, the approach is still immature and should be investigated when different data sizes, data distributions and workers are handled while ensuring the model performances[4]. These challenges hinder FL from being adopted by various industries[11]. Therefore, the existing frameworks are at very early stage of development. Due to the current open issues, Lo et al. highlights that FL is still under-explored and the main motivations for using it are also the most studied challenges.

The review from Lo et al. also reflects that, due to the current open issues, FL is still under-explored and highlights the main motivations for using FL in the existing and state-of-the-art approaches. Those motivations appear to be also the most studied research challenges.

To address them, a scalable and extensible FL framework for a practical usage in the healthcare domain should be designed. In the ODIN project, the objective is to provide a flexible (componentized), extensible, secure and scalable in processing resources FL framework that can be reconfigured and adapted to the ranges of reference use cases (RUCs), addressing the healthcare needs of pilots.

The proposed FL pipeline should be customized to other health domains that leverage the AI resources in the ODIN platform, so that different sites with their own data can collaborate by training the same model without sharing private information. This leads to build a collective knowledge and accelerate the implementation of AI in operational environments.

2.2 Data

Regarding that some of the AI models used for emergency prediction and handling are raw images and their labels, it is beneficial to set up a way to share the improvements on those models. Since the models, such as the ones evaluated in *Figure 5: Object detection State of the art algorithms comparison* are usually trained with open-source data for comparison purposes; for application purposes there exists the possibility to fine-tune the models with real labelled data. With this scope, the federated learning can benefit each pilot with the other hospital's data, that can be used to increase the amount of data to perform the fine-tunning.

---

[10] J. Mills, J. Hu, G. Min. 'Communication-Efficient Federated Learning for Wireless Edge Intelligence in IoT', IEEE Internet of Things Journal, 1–1, 2019.

[11] L. U. Khan, W. Saad, Z. Han, E. Hossain, C. S. Hong, 'Federated learning for internet of things: Recent advances, taxonomy, and open challenges' IEEE Communications Surveys & Tutorials.

3 Emergency definition

To clarify the motivation of the deliverable, a section is dedicated to describing the needs and situations to handle that will be covered in this document. The description of the emergencies listed below corresponds with the emergencies that are requested to be solved by ODIN by the pilots involved in the Reference Use Case C, which aim is to improve the disaster preparedness of the hospitals and improve the immediate response on disasters that are susceptible to happen in the environment of a hospital.

Although the scope of the emergencies defined is also limited by the range of possibilities of the AI that is being developed in the ODIN project and is not a one for all solution. That is why we purpose a resilience system inspired on the one designed by Hendrickx q et al. in which future solutions to emergencies that are not listed below are evaluated to be solved by the AI in the following release of the task 6.2.

3.1 Types of emergencies

a. Disease transmission

Hospitals are the first place to be susceptible of being a focus on disease transmission that's why it should be the first place to be protected against them. It is true that during COVID-19 pandemic hospitals where, despite all, one of the most organized places against the transmission of the virus but even though it is not easy to control, avoid and track bad practices that can have consequences inside and outside of the hospital.

b. Chaotic evacuations

It is believed that hospitals whilst being public places with very unpredictable occupation are susceptible to surpass the maximum occupation of the building, room or floor. It is suggested a solution to palliate that risk for the hospital to be a safer place or at least to detect the risk situations that are until now very difficult to realise.

c. Forgotten objects

Potential disasters can come from forgotten objects that most commonly will mean no harm at all but there have happened cases in which a forgotten object caused drastic injuries.

d. Valuable asset localization

It is considered an emergency the event where a valuable item, necessary in an emergency or in any other case cannot be found. For this purpose in this deliverable, it is provided a use case in which a localization system is applied in this objects.

e. Patient falling

In some cases, a fell can cause severe consequences in patients and a fast response is crucial and sometimes it is possible that there is no member of the staff around that notices the incident.

3.2 Tiered system

Since not all the emergencies are equally dangerous it is suggested a system in which the emergencies detected are classified according to the urgency of the matter, but also to the probability that they happen. It is necessary then to identify and classify all the risks that the ODIN platform can detect and assign a group of people that will deal with the problem.

To that interest it is purposed the following *Table 2: List of possible emergencies* to evaluate the risks:

Table 2: List of possible emergencies

Emergency level	Probabililty	Risk evaluated	People attained	Measures taken
Very high	Very low	Fire collapse	All	Fire detection
Very high	Very low	Evacuation collapse	All	People counting
Medium	Medium	Maximum Capacity	Management/Security	Restrict entries
Medium	Yearly	Epidemic	Nursing	Mask detection
Medium	Medium	Falling risk	Nursing/closest	Fall detection
Very low	High	Forgotten object	Security staff	Object detection

Even though this is a possible full alarm system table, it is expected from the pilots to fulfil a table with their own risks and alarms. The aim of the next version of this deliverable, is to narrow down the possibilities described in this deliverable to a pilot scale.

3.2.1 Involved staff

For the system to be effective, it is important to clarify which people (nursing, security, management…) will be in charge to take responsibility on the actions to resolve the problems detected. So, it will be crucial to clearly inform the people involved how to proceed, and what they need to know to solve the problems detected.

It is purposed to set up a way to reach different teams of staff, for example via cell phone (slack, teams, SMS…), the different groups of people will be noticed when there is an event detection that requires their assistance.

Although this document focuses on what actions can be detected by the ODIN platform and the definition of the measures to be taken lies in the hospital itself since each case will have particular factors and a previous organization.

3.3 Resilience system

Once defined the emergencies it is important to define the actions taken to solve the problems but also to find the causes and evaluate the possibility to fix the cause for future occasions. In the case of ODIN an alarm system is purposed to solve some of the problems detected. The functionalities and possibilities of the system are listed below.

a. Object detection

Further than people detection, object detection has a possible application to prevent obstructed corridors, forgotten objects or any other kind of suspicious package that might be a potential problem, for example in a building evacuation.

The newest algorithms trained with the most powerful databases existing will be used to this task, which by now is the newest version of YOLOv7. Which also includes the possibility body tracking, person reidentification (without biometric identification) and object segmentation. Also the Mysphera IoT systems will be applied with the scope of tracking valuable property of the hospital.

b. Crowd management

Crowd management is an important issue to prevent not only an emergency situation in which a crowded room had to be evacuated but also to manage the customers in order to avoid long queues and other applications such as:

- Waiting time reduction

A possible problem in crowded hospitals would be how to deal with the people that would come in. A computer vision system would be able to know how to distribute them in case it had multiple queues or waiting rooms.

- Access control

It is possible to detect the amount of people that enter in a room, in case only a limited number of people are allowed, or to control restricted areas via identification (such as a specific clothing of the hospital staff).
It is also appliable to unrestricted areas just to take into account how many people enter in a room, e.g. number of familiars visiting a patient, number of people that have been in the toilet, etc.

- Rise of temperature prediction

Since a big amount of people may infer in the temperature of a room, detecting an overcrowded scene can be used to prevent a rise in temperature and the according measures can be taken.

- Automatic lighting control

Currently a lot of rooms count with a sensor that detects movement, but with crowd control it is possible to take into account not the movement but the presence itself of a person in a room that would avoid shutting down the lights if no movement is detected.

c. Fall detection

The fall detection algorithm developed itself by INETUM is planned to detect and help accidents where someone falling is detected. It consists in a deep learning model that considers the body tracking of the persons detected in camera and as an output has 5 possible labels that will say whether the person detected is Standing, Sitting, Lying, Walking or Fallen.

d. Nosocomial analysis

Disease transmission is one of the most relevant concerns in a hospital, with the aid of AI and IoT it is intended to mitigate and track the events related to disease transmission. With MYSFERA ATLAS solution the ODIN platform will be able to extract and analyse relevant information about it. See page *16* for more information.

4 Emergency prediction systems

4.1 Computer vision assisted solutions

To identify possible emergencies a computer vision assisted system is going to be developed. The system will include a network of cameras that will be continuously identifying possible emergencies that can be visually identified in a hospital. Although, since there exists still the possibility for the algorithms to alarm for a false positive a method to confirm the alarms will be set if needed,

4.1.1 Architecture of the camera network

An architecture is purposed where all the cameras are connected to a central server where the images are sent to be processed in the cloud or in premises, the alerts are then triggered and sent to the corresponding staff.

The video streams gathered by a set of cameras and/or sensors will be collected and managed simply with a streaming platform such as Kurento, Janus or a similar platform. Will then parallelly be sent to the server by RTSP where they will be processed by the computer vision algorithms. The output data of the computer vision algorithms will be sent to the ODIN platform, that can be in the cloud, where it will be processed to perform predictions on peak hosting, COVID control, and will also trigger alarms at the accomplishment of different conditions when emergencies are detected.

The alarms will be sent to the ODIN platform via Kafka message to a specific topic for emergencies that will trigger messages to terminal devices of the corresponding staff needed in the emergency scenario.

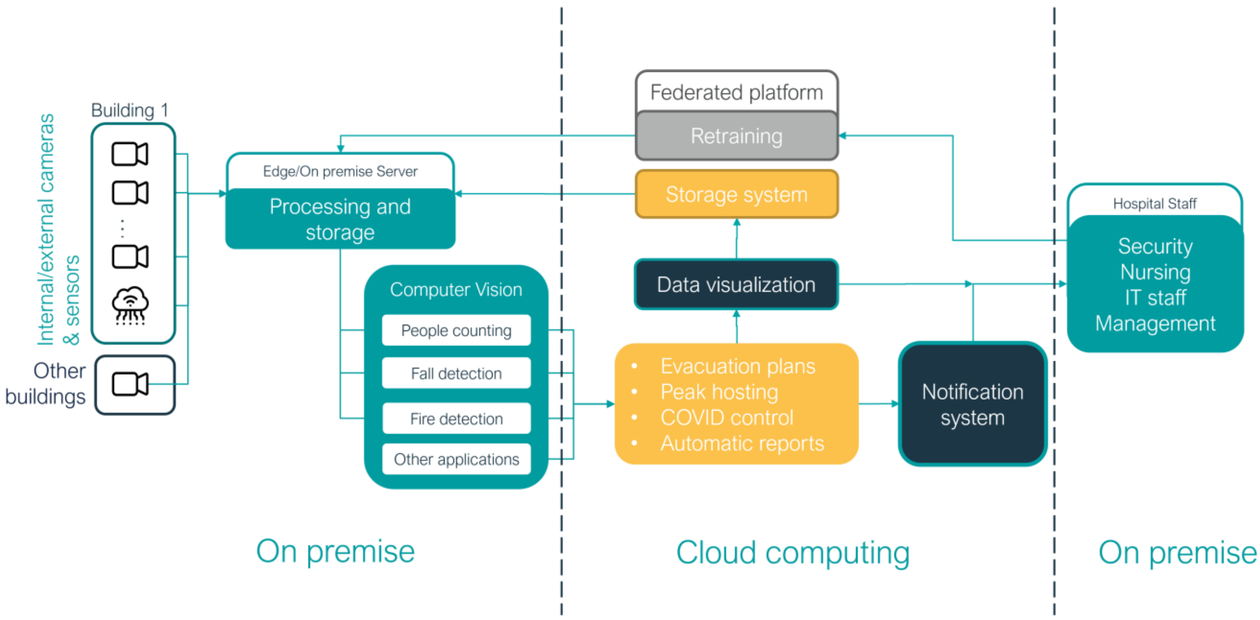

Figure 3: Computer vision system pipeline

The gathered data will also be stored to perform statistics to detect and predict relevant data that will be used to improve the resources dedicated to managing the availability of the staff.

There is also a need of a channel in which the alerts will be deployed which will be agreed with the pilot hospitals, a PoC is developed in which the alerts are sent to chat in the Slack platform but could be any other platform.

4.1.2 The processing

The latest state of the art algorithms at the time will be used to satisfy as best as possible our goal, to prevent any predictable disaster seen by a camera. These algorithms are constantly being improved by the community and considerable improvements can be made in the future. For example there exists already the next version of YOLO, the YOLOv8, which does not give a significant improvement in accuracy nor speed, but in usability, and documentation.

Thus the whole processing is mainly divided in 5 main parts consisting on the video grabbing, image pre-processing, the object detection part, which uses the YOLOv7[12] architecture model trained with the MS COCO database, the object thresholding, that filters the false positive detections and object tracking, which assigns an id to every detection made and relates every detection in close positions between frames, this way we can tell what is the track of a detection, or how long a detection has been on screen.

For the object detection it has been chosen a "You Only Look Once" type of architecture YOLO is an architecture of a one-shot object detection, it means that the image is only inferenced once through the model, this type of object detectors is usually faster, more suitable for real time applications, in exchange of some accuracy with respect to other two-stage object detectors.

Compared to other one-shot detection algorithms such as faster-RCNN and other predictors that use a Region Proposal Network, the architecture of the YOLO algorithms is faster as it does not work by detecting possible regions of interest but with one only fully connected layer that performs all the predictions.

It is also chosen because of the tools that brings with it, that can be very useful in case other applications are needed. Such as object segmentation or pose estimation.

From the processing point of view, we define the pipeline with the following processes that are described below.

1. Video grabbing

From the IP cameras disposed in the building each one is connected via real-time streaming protocol (RTSP) to a processor with a defined specifications (see 4.1.5 alg. performance/ hardware specs). The image stream is grabbed with full quality from a server or processor that can be allocated on premises or on the cloud.

2. Video pre-processing.

The image stream is passed to an iteration and each image is processed independently, so the processing is made once for each image grabbed from the stream. Once the image is in the iteration it is pre-processed for the processing to be made in a reasonable time. A normalization and redimensioning process is needed. Once those processes are carried each image is ready for the inference process.

[12] Chien-Yao Wang, Alexey Bochkovskiy, Hong-Yuan Mark Liao YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors https://github.com/WongKinYiu/yolov7

3. Object detection

The object detection process is carried with the YOLOv7 type of architecture, this type of architecture is one of the most used and efficient object detection models of the state of the art.

The backbone architecture is called E-ELAN, which stands for Extended Efficient Layer Aggregation Network. This network backbone is designed with the aim to improve the speed and the accuracy considering the memory access cost, input/output channel ratio, element wise operations, activations and gradient paths.

In simple terms, the E-ELAN architecture allows the framework to learn better and get more robust results without sacrificing memory allocation, not time per batch.

Thanks to the model rescaling feature of YOLOv7 it is possible to rescale models to adapt them to different applications, this means that once a model is trained it can be rescaled to improve the model in terms of time spent per frame processing.

In the case of the RUC C the YOLOv7-W6 is chosen due to the great balance between FPS-accuracy. In the YOLOv7 paper it is said that this model can reach up to 84 FPS and 72.6% of accuracy. Which is a very big improvement in processing time with respect to previous detectors, as it can be seen in *Figure 5: Object detection State of the art algorithms comparison*, which

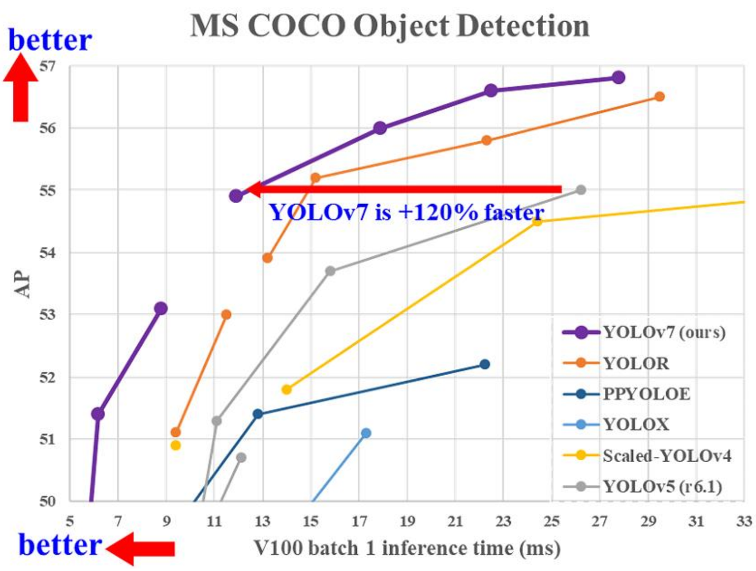

Figure 5: Object detection State of the art algorithms comparison

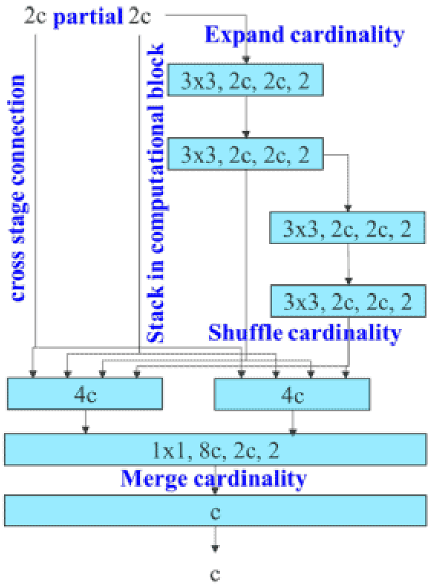

Figure 4: YOLOv7 backbone architecture

compares different models of other similar YOLO algorithms.

4. Detection thresholding

Once the detections are made a crucial method to improve accuracy is to get rid of the false positives applying the non-maximum suppression (NMS) method. This method is needed to get rid of all the repeated detections. Conveniently it is very useful to dimensionate the algorithm, since the most detections the slower it will be the algorithm, not because of the detection step, but because of the processing applied to each detection. Since NMS needs a threshold for the Intersection over Union (IoU) from which it will discard the non-maximum intersected detections, we can lower this value to get more detections, sacrificing processing time, or increase it, making the algorithm faster but less precise. Based on a testing experience in the real environment this parameter can be optimised in order for the process not to waste a handful of resources. It is by default left in a relatively low value, so it does not supress any real detections.

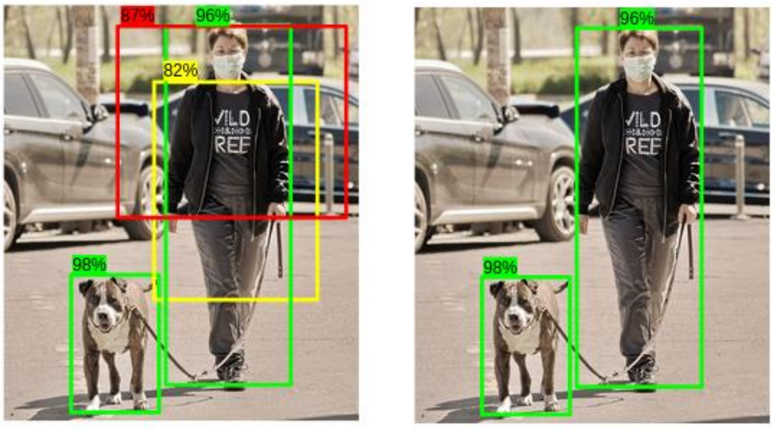

Figure 6: NMS using IoU example

It is also allowed to filter the output detections by classes, so the output consists only of the needed classes, such as person, handbag…

5. Detection tracking

The object detection and classification are done all in the same model and it is a very fast step of the system. It is though done frame by frame independently and thus some of the needed features cannot be implemented such as time elapsed since first detection. To do so the *deepSORT*[13] model is needed.

Tracking a detection means to compare and associate detections between frames, this means that we are already working with more than one frame, which allows us to add a time dimension to our output data. This comparison is added in regard of the position of the detection and also of the distance between the content of each detection, in a non-deep solution this distance would be calculated for example comparing histogram statistical measures, but in our case, with the aim to accelerate the process it has been decided to use a metric based on deep learning. By now the *OSNet* model for reidentification is being used, but this could be improved in a future.

The algorithm also includes other methods such as Kalman filter to avoid the loss of the track due to occlusions.

Since this algorithm is constantly comparing detections between images, and applying several other processes, the more detections, the slower the algorithm will be. So, for instance, the average frame rate is around 15fps, which is considerably slower than before, although it is still considered real time and it is feasible for the purposes of the use case.

4.1.3 Semantic Mapping

Since the ODIN platform takes advantage of this information it is also important to keep a track of the room, and the camera identification number in which every detection has been made as well as any other relevant information such as angle of the camera or the aperture of the objective,

[13] Nicolai Wojke, Alex Bewley, Dietrich Paulus SIMPLE ONLINE AND REALTIME TRACKING WITH A DEEP ASSOCIATION METRIC

directly related to the angle of vision. With this information every detection can also be related to a room name, so it is recognizable by the eWorkers. This way the whole ODIN platform can take advantage of this information which is very valuable for example for the robots which they already use semantic mapping.

Thus, it comes up the possibility to merge solutions in order to enhance the real time database available that the robots have. This possibility is to be evaluated by each pilot and in case it is of interest could be developed in collaboration with WP5.

Another benefit of keeping track of the semantic mapping is to perform periodic reports, as mentioned in D6.2 of occupation per room, per floor, and building if there is more than one.

4.1.4 Further processing

As said before, YOLOv7 repository includes a very powerful tool of pose estimation and since there is a possible tool to be applied it has been developed with the intention of testing its possibilities. This algorithm has been used to train a model of fall detection that can be used to detect fall incidents or to track someone's activity, the results of this module can be drawn with understandability purposes.

The fall detection model is trained with the keypoints extracted with the pose estimation method. Each set of keypoints behaviour is associated to an action, so the output of this model is an integer number 1 to 5 associated with a concrete action.

The model is called TSSTG[14] and is a model designed for action recognition purposes. It basically needs the data of the pose estimation of 30 frames to get to an output for a human detection, the model itself is not very expensive since it is only trained with data points, although still a small convolutional neural network (CNN) is built and ran over 30 frames at once.

The repository implementing the paper claims to get up to 7fps, but the truth is that in real life with 10 detections at the same time, the frame rate is below 1fps, this means that this algorithm will barely work for real time applications, and is probably not suitable for the RUC C.

4.1.5 The Fine tunning

To continuously improve the performance of the algorithm and respecting the GDPR of the hospital a fine-tuning system will be set up in which the hospital staff will be able to do all the labelling and processing of the images with a minimal guidance so no one outside of the hospital will see those images.

4.1.6 Hardware specifications

The current system is being tested in an Amazon Web Services server and it can run with a GPU NVIDIA T4, with 16GB of memory and 65TFlops we can run the tracking model at 15fps. More tests will be made in a close future to recommend the best hardware for the pilots. The cameras used to test the algorithm were Avigilon H5A.

There are several ways to approach the processing of the computer vision system that need to be evaluated by each pilot considering different factors such as the current infrastructure, the

[14] Sijie Yan, Yuanjun Xiong, Dahua Lin. Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action

budget that they are willing to deploy the system and the functionalities that they are willing to implement.

- Edge computing can be performed to run the algorithms. A specialized hardware is usually employed to that matter, mainly consisting on-edge computing built in machines, such as the NVIDIA JETSON series. These processors allow to execute inference on light AI CNN models of multiple cameras at the same time.

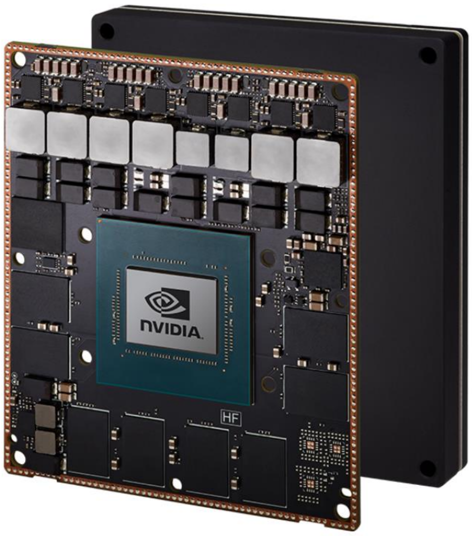

Figure 7: Nvidia Jetson Xavier

- Centralized server. This option would need a considerable power of computation to execute the multiple video streams in parallel, which could imply a considerable inversion. The benefit of this option is that the server is easily resizable and could be upgraded when needed.
- In camera processing is an option since many camera manufacturers include nowadays camera AI computing which would implement some basic applications of object detection. This is a feasible option but is not resizable since the algorithms depend only on the camera manufacturer and the accessibility to the data is completely unknown. This option would also need a considerable inversion in specific camera models, in a server sold by the camera company to decode the results of the AI as well as a periodic fee for the cloud services of the provider.
- A last option would be to send the image stream to a cloud computing server and perform all the computations there. Since the images are considered sensible data of the hospital it is usually not feasible to take the images out of the hospital, so this option would be in most cases discarded.

4.2 Sensor assisted solutions

Sensors have been used in Emergency Management (EM) systems in a wide variety of tasks, since they were introduced, but it was not until the decade of 1990, when the United Nations General Assembly designated 1990 as the International Decade for Natural Disaster Reduction when they started to be introduced more in depth and the EM were developed.

As pointed by Di Huan et alters in "A systematic review of prediction methods for emergency management"[15] , sensors have been used in a wide range of tasks in the EM cycle, starting in the pre-event stage or prediction tasks, following the In-Event stage when an emergency is happening and finally in the Post-event stage when a post-mortem analysis and a recovery from the disaster must be performed.

Usually, the sensors used in EM can be grouped into Remote sensing, Geography Information Systems (GIS), and Global Positioning Systems. The combination of three, provide environmental, spatial, and temporal information to feed an EM.

For the purpose and interest of this project, it can be inferred for a hospital that the EM can also have those groups of sensors or systems. In this case the analogy would be also Remote sensing for environmental and general sensors, Building information System (BIM) instead of GIS and Real time location System (RTLS) instead of GPS.

Usually in EM, Remote sensors are related to light systems, Fire preventing systems, Water pipe systems, Radioactivity sensors, Electrical failure systems and HVAC systems (Heating – Ventilation - Air conditioning). Those systems have proprietary interfaces and its deployment go beyond the scope of ODIN project; thus they will not be tackled deeply, but in case hospitals want to integrate, they are welcome. On the other hand, ODIN will provide robots with sensors, that can measure temperature, noise and air quality, which may also help to prevent disasters.

BIM systems are another example of closed solution that is out of the scope of ODIN project, but ODIN will have its own Resource Manager and together with positions and maps would do the job of a BIM partially.

Finally, RTLS systems are part of ODIN project, and the following sections will review how an RTLS can help on EM.

4.2.1 Architecture of the RTLS

The RTLS is composed by several items. The base infrastructure are Bluetooth gateways deployed into the hospital with the aim to cover all the areas under interest with a good signal to noise ration. The gateways, listen to Bluetooth Low Energy (BLE) tags, which send radio signals at a given frequency. The tags can be attached to high value assets (such as robots, medical equipment, etc) or patients and medical staff to track them.

[15] Di Huan et altres. A systematic review of prediction methods for emergency management. https://doi.org/10.1016/j.ijdrr.2021.102412

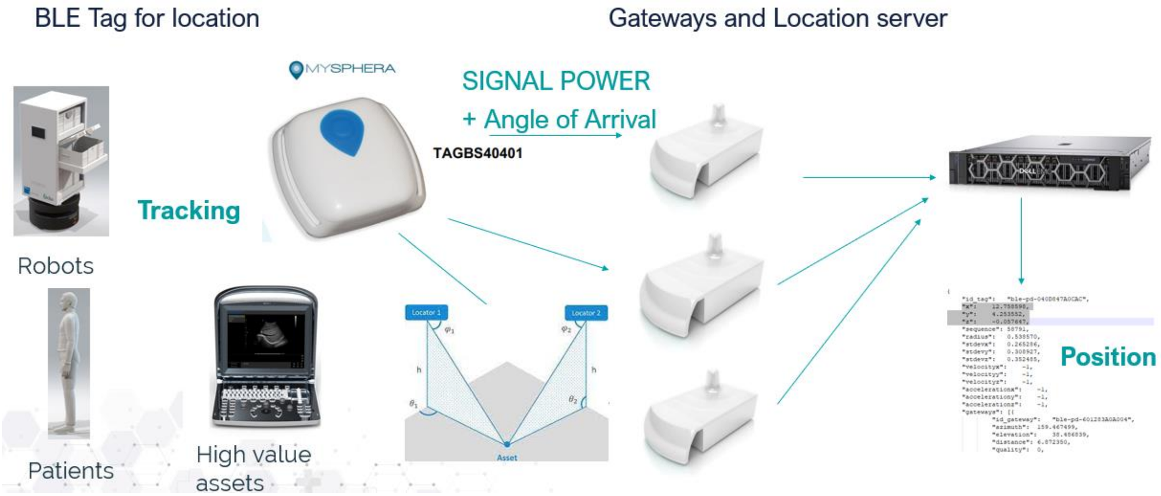

Figure 8: RTLS components

The gateways send the signal received from the tags to the location server which computes the position of a given asset or person using and algorithm called Angle of Arrival and publishes it to the ODIN Kafka bus.

With this information, the system can store and process the data using AI functions located in other services, to perform predictions, analysis and other tasks for EM.

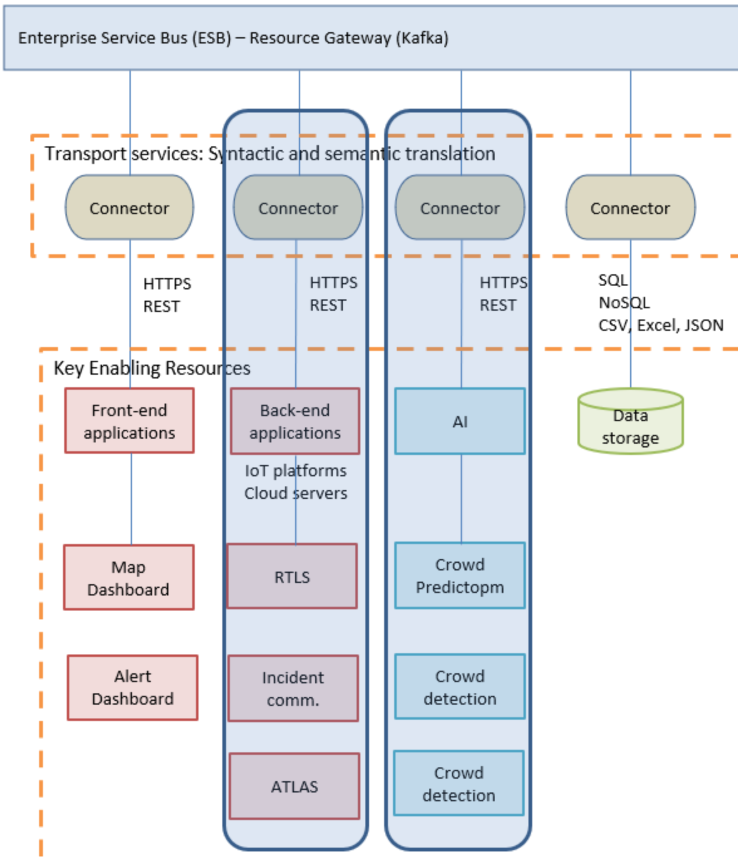

Figure 9: RTLS and AI in ODIN Architecture

Figure 9: RTLS and AI in ODIN Architecture*9 RTLS and AI in ODIN Architecture* shows how RTLS publishes asset and people position to the Kafka bus. This information can be read by AI functions, such as (Crowd prediction, and other types of AI focused on other problems) or stored into SLQ or non-SQL databases for training models or batch processing. Other applications such as displaying Dashboard Maps, the ATLAS solution or Alert Dashboards can work also reading and processing information from the RTLS or the results of an AI processing.

4.2.2 The processing

There exist several use cases that can be applied to help on EM predictions and handling with an RTLS.

a) Patient, medical staff or high value asset location during a disaster, emergency, or evacuation:
EM can receive a very valuable information about where are located the people and the high value assets so it can be planned actions during a disaster. For example, for evacuation or relocation of people and equipment.
Dashboards highlighting people and assets help detecting areas to be evacuated.

b) Robot routing for evacuation:
Robots are semiautonomous systems and upon an emergency can receive a route to evacuate themselves to a safe location or take actions to help evacuating people in case they are monitored using an RTLS.
In this case an AI can compute the route a robot may follow upon a disaster.

c) Nosocomial infection spread:
With an RTLS in can be analysed the contacts an infected person has had in the previous hours. This is what MYSPHERA ATLAS solution provides, making easier for the hospital's Biological Control Unit to mitigate the infection spread and find and isolate the possible infected people. Infections such as COVID-19, pneumonia and others with high contagious rate can be controlled better with this solution.
In this case, the models work on the basis that close contacts among the patients and staff can happen in a given radius. Length of the contact is also critical to highlight possible contacts infected.

d) Resource and staff availability:
In some cases, an area or service may run out of some assets or staff for a while, which under some circumstances can lead to a disaster or at least decrease the efficiency of a service. With an RTLS the EM can notice there is missing wards, medical staff or assets like ultrasounds, defibrillators, food pumps or other equipment that could generate complication in case an emergency occurs. For example, in case a defibrillator is missing, the probability of mortality of a patient fibrillating, increases dramatically.

e) High value asset stealing:
Controlling high value assets is difficult in some cases where open doors are available, and guards are not present all the time next to the entrance doors. Automatic alarms upon detecting assets going outdoors help staff to control thieves.

f) Crowd prediction and detection:
RTLS can help predicting and detecting crowds, but that means that everyone in the hospital or at least a critical mass of people is monitored with the RTLS. AI can take

position and directions of the traces of assets and people to predict where they are going to. Also, the RTLS can help predicting bottlenecks and detecting areas such as emergency waiting rooms, corridors or other locations that can be impacted by the presence of crowds.

g) Incident communication:
The RTLS smart bands have buttons that can be programmed to trigger actions, thus situations such as a patient who needs help or reporting an emergency (Fire, Flood, etc) can be programmed into the smart band's button.

5 Emergency handling systems

5.1 Notification systems

The aim of the system is to warn the people involved in the solution of the emergency including the staff such as security personnel, nursing, etc. to do so we need to establish a channel of communication between the relevant eWorkers and the ODIN EM system.

From the ODIN platform the data processed will trigger several alarms through Kafka that will be linked to the hospital system. The conditions that must accomplish to the alarms to trigger are to be set by the hospital itself, since they can be associated to parameters such as the maximum capacity of room, floor or building, which will be different for each pilot.

Even though the parameters might not be the same for each hospital, given the data that we can gather with the system here is a *Table 3: List of possible notifications* with all the alarms that can be triggered through the system.

Table 3: List of possible notifications

Notification	Hospital Data needed	Confirmation needed	Receiver
Max. building capacity reached	Yes	No	Security
Max. floor capacity reached	Yes	No	Security
Unmasked person detected	No	No	Security
Fire detected	No	Yes	Everyone
Fall detected	No	Yes	Nursing
Object forgotten	No	Yes	Security
Valuable asset not found	Yes	Yes	Security
Rare people flow detected	Yes	No	Management

Every alarm will be sent to a different group of people for which a different Kafka topic will be created. Again, the hospital is responsible to choose which group of people is responsible to solve the incident detected, but similarly to Table 2, in the Table 3 an example of different groups of people have been estimated.

Some of the alarms could be in trouble with the GDPR rules and that is why it will be carefully considered in from the processing until the notification process which features are included in the system.

Since the alarms are set by a computer it will be necessary sometimes to check the notification by someone, it is possible to send a picture with the alarm in order to make easier the confirmation, or it can simply be checked by looking at the video stream where it was detected.

5.2 Automatic reports

Given that the crowd flows in the hospital are registered with a certain accuracy, it is possible to put all this data in a visual way for the management staff to have a clear vision and understanding of the situation on the hospital.

To that matter it is suggested that periodic reports are available, for example with a periodicity of the range of months, in which the occupancy of the building, floor, or even room are available.

Also, once enough data is gathered it would be interesting to perform data analysis techniques on the information available. If there are patterns that guide this data then machine learning can be applied to measure them, prevent them or even relate them to particular contextual events.

As explained in deliverable D6.2 the reports are adaptable to the hospital necessities and will be built hand by hand with pilot advisors considering their needs and preferences. Anyhow, the possibilities to visualize the data are wide and a particular time interval and space will have to be chosen for the sake of simplicity.

6 Conclusion and next steps

In this deliverable several available tools and methodologies that the ODIN project can reach to build an AI enhanced emergency handling system have been described. It has been reviewed how an emergency system works and how an AI system could fit in the ODIN project, taking advantages of a federated learning platform which would improve the models used in all pilots.

Mainly, two solutions have been proposed for emergency detection. The first, refers to computer vision to control the crowd flows and detect any possible causes of a potential disaster such as people not wearing a mask, overcrowded scenes. The second consists of the application of RTLS devices with emergency detection purposes. Some of the benefits of this proposal are the tracking of the contacts of infected people and emergency alerting between others.

The emergency handling system is based on alerts that will be triggered after analysing the data gathered by the methods used. These alarms will have to be related to the relevant eWorkers that will be responsible to take any actions on how to solve the detected situation.

These systems are usually focused on handling or preventing an emergency situation. What it is intended in the future, is to use AI enhanced technologies, not only to prevent but to predict this kind of situations. With this aim, several proposals motivated by other pilots are over the table.

Even though the algorithms developed are still to be tested in real conditions to verify their accuracy and evaluate the benefits of a retraining process there are already some other extensions being considered. Some of them are considered even though the data collected by the system already described is needed to test and build them.

On the next period we will focus on the development of the systems planned and the proposals of the pilots to complete the use case of disaster preparedness. Since in this deliverable the processes and tasks have been described, it is expected that in the following deliverable it will be reported how the pilots are performing the tasks and procedures described above. Some of the tasks to be considered in the next steps are the following:

– Test the developed algorithms and embed in the notification system and the storage system to start creating the database on crowd flows in the building.
– Perform data processing on the output data of the current system. This will allow, not only to detect hosting peaks but to prevent them by identifying patterns of people access the hospital and when. To do this, classical machine learning methods can be applied such as k-means and other clustering techniques, polynomial regression, Fourier transform analysis to detect periodicities.
– Use hospital and public data on diseases to perform epidemiologic data models. State of the art epidemiologic models such as SIR, as well as other more innovative solutions can be used to perform epidemiologic simulations.
– Still other solutions are over the table such as context analysis through scientifical articles, news, social papers and other possibilities.
– Integration in the ODIN platform and the systems that conform the platform.