UNIVERSITÀ DEGLI STUDI DI CATANIA
Dipartimento di Matematica e Informatica
Dottorato di Ricerca in Informatica XXXIII Ciclo

Francesco Ragusa

Human Behaviour Understanding from First Person (Egocentric) Vision

Tesi di Dottorato di Ricerca

Supervisor: Prof. Giovanni Maria Farinella
Co-supervisor: Dott. Antonino Furnari
Company Tutor: Ing. Emanuele Ragusa

Anno Accademico 2020 - 2021

"*Only those who will risk going too far can possibly find out how far one can go.*"

T. S. Eliot

Abstract

The First Person (Egocentric) Vision (FPV) paradigm allows an intelligent system to observe the scene from the point of view of the agent which is equipped with a camera. Wearable cameras allow to collect images and videos from the humans' perspective which can be processed using Computer Vision and Machine Learning to enable an automated analysis of humans' behavior. To study the human behavior from the first person point of view we considered both cultural heritage and industrial domains. Equipping visitors of a cultural site with a wearable device allows to easily collect information about their preferences which can be exploited to improve the fruition of cultural goods with augmented reality. The inferred information can be used both online to assist the visitor and offline to support the manager of the site. Despite the positive impact such technologies can have in cultural heritage, the topic is currently understudied due to the limited number of public datasets suitable to study the considered problems. To address this issue, we proposed two egocentric datasets for visitors' behavior understanding in cultural sites. Together with the datasets, we proposed 5 fundamental tasks related to visitor behavior understanding, which can be addressed using the proposed datasets. Moving from these studies, we built the *VEDI System*, which is the final integrated wearable system developed to assist the visitors of cultural sites. While human-object interactions have been thoroughly investigated in third person vision, the problem has been understudied in egocentric settings and in industrial scenarios. To fill this gap, we present MECCANO, the first dataset of egocentric videos composed of multimodal data to study human-object interactions in industrial-like settings. The multimodality is characterized by the gaze signal, depth maps and RGB videos acquired simultaneously with three different devices. The dataset has been explicitly labeled for the tasks of recognizing and anticipating human-object interactions from an egocentric perspective. We report a benchmark aimed to study egocentric human-object interactions in industrial-like domains which shows that the current state-of-the-art approaches achieve limited performance on this challenging dataset.

Contents

Chapter 1

Introduction

1.1 Motivation

The ability to interpret the surrounding world allows humans to interact with other people, localize themselves in an environment as well as manipulate and grasp the objects present in the scene where they live and work. The aforementioned ability, called *visual perception*, is given by the human visual system, and it is specifically related to the cognitive abilities of the brain to understand the observed scene given the image formed from the eyes. This concept has inspired the Computer Vision field and its research community emulate human visual perception system with an artificial system. Nowadays, with the diffusion of optical devices (e.g. digital cameras, surveillance cameras, phone cameras, etc.) which are able to simulate the human eyes to acquire a digital visual representation of the real world, the computer vision community is able to develop intelligent systems which answer questions about the acquired scene.

These systems are part of our daily life, for example in the shops where fixed-cameras count people entering or leaving the store or when you take a photo of the food with your smartphone which is then automatically described and organized. As additional examples, some companies have installed security systems which automatically authenticate the users analyzing their faces whereas in the autonomous driving field, there are key features comprising the use of an intelligent systems based on cameras which are installed on the car to understand the scene, detect pedestrians or road signs. Nowadays, first person vision systems are used to assist and improve the quality of life of humans with disabilities. For instance Orcam[1]

[1] https://www.orcam.com/

developed wearable intelligent systems able to improve the lives of individuals who visually impaired.

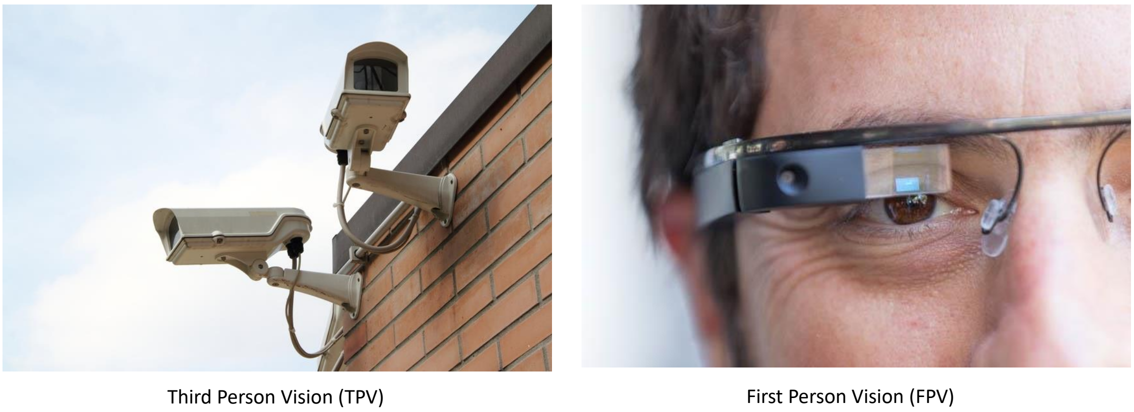
The artificial intelligent systems will be ever more present in our lives due to the growing effort and interest of the research community and the positive effects which they have on our lives.

1.2 First Person (Egocentric) Vision vs. Third Person Vision

The Third Person Vision (TPV) paradigm assumes that the scene is acquired from a fixed-camera which is neutral with respect to the observed environment (see Figure 1.1-left). With this paradigm, many Computer Vision problems have been addressed by researchers, such as face recognition, object tracking, scene understanding, etc. These cameras present physical constraints related to their fixed position, but allow to solve specific problems such as surveillance of a specific area or providing parking services to look for empty parking spaces. The surrounding environment changes during the interactions with the agent (i.e. the position of the agent, the state of the objects he has manipulated or the persons which are present in the scene). The agent could be a human, a robot, etc. In the literature, these dynamic factors are defined by the concept of *context* [1, 2].

Differently from TPV, the First Person (Egocentric) Vision (FPV) paradigm allows an intelligent system to observe the scene from the point of view of the agent which is equipped with a camera (see Figure 1.1-right). In this way, the intelligent system represents a dynamic agent which can observe and interact with the surrounding world. A wearable system can assist a human agent to understand how to interact with a specific object or it can localize the human in the real world to suggest where to go.

The main difference between the TPV and FPV paradigms is related to the mobility of the camera. Fixed cameras have a static point of the world while a wearable camera is placed on the agent and its position is dynamic. The dynamic movement of the wearable camera introduces some artifacts like motion blur, otherwise, the captured information of the scene is meaningful to understand what the agent is looking at and what are his intentions [3]. A FPV system is developed to perform



Third Person Vision (TPV)

First Person Vision (FPV)

Figure 1.1: Example of fixed-camera representing a Third Person Vision (TPV) system (left) and of wearable camera usually used in First Person Vision (FPV) systems (right).

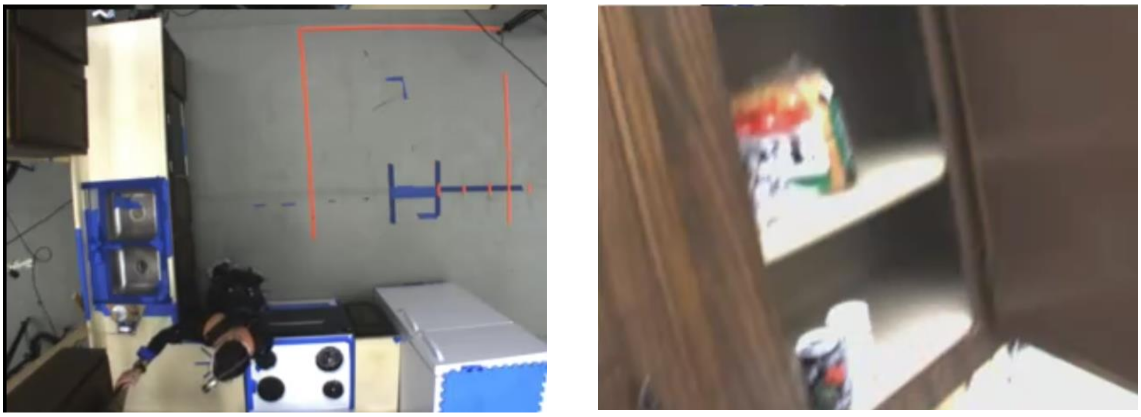
multiple tasks considering the dynamic nature of the observed scene. An example regarding the difference between the two paradigms is reported in Figure 1.2. The figure compares two images of the same environment acquired from two different points of view (i.e., TPV (right) and FPV (left)) in which the human is opening a door of the cabinet. From the first person point of view, some information are more representative regarding the intention of the human (e.g. which objects in the cabinet the human is looking at?). In this thesis, the FPV paradigm is considered to understand the humans where they live and work.

1.2.1 Wearable Devices

Wearable devices have been studied since the 80s. Steve Mann developed and built the first wearable computers [4, 5]. These devices were able to capture images of the surrounding environment and return some feedback thanks to a display. The wearable computers developed by Mann represent the first prototype systems of first person vision. Figure 1.3 shows the temporal evolution of the wearable computers designed by Mann.

Nowadays, the development of wearable devices is constantly growing. An example is given by the action cameras used in the field of sport (i.e. GoPro[2]). These wearable cameras (see Figure 1.4 - middle) often have small dimension and have

[2] https://gopro.com/it/it/



Frame from Third Person Camera

Frame from First Person Camera

Figure 1.2: Difference between two images which captured the same scene from two different points of view. On the left is reported the image captured from the TPV camera, while on the right the one captured with a FPV camera.

Table 1.1: The main wearable devices present in the market with some details.

Device	CPU	GPU	RAM	Camera	Technolgies	Price
HoloLens	Intel 32 bit	Intel HPU	2 GB	2.4 MP, HD video	Mixed Reality	3000 $
HoloLens2	SoC Qualcomm Snapdragon 850	Intel HPU 2th-gen	4 GB	8 MP, video 1080p 30 fps	Mixed Reality	3500 $
Google Glass	TI OMAP4430	PowerVR SGX540	0.682 GB	5 MP, video 1280x720 px	Augmented Reality	999 $
Vuzix Blase Series	4-core ARM A53	N/A	1 GB	8 MP, video 720p 30 fps, 1080p 24 fps	Augmented Reality	1085 $
Magic Leap One	6-core	NVIDIA 256 Cuda Graphics	8 GB	2 MP, video 1080p 30 fps	Augmented Reality	2730 $
Tobii Pro Glasses 2	N/A	N/A	N/A	1920x1080 25 fps	Eye tracking	11900 $
Pupil Invisible	N/A	N/A	N/A	1088x1080 30 fps	Eye tracking	6500 $

fewer interaction elements (e.g. buttons or sticks). Many companies have invested in the development of wearable devices (i.e. wearable glasses). For example, Microsoft developed the HoloLens device[3], which allows the use of mixed reality to assist the users in an environment (see Figure 1.4 - left). Tobii built a wearable glass, called Tobii Pro Glass 2[4], which is able to track the gaze of the pupils (see Figure 1.4 - right). The aim is to acquire useful data to understand how a human interacts with the surrounding environments and what catches his attention. Table 1.1 reports the main wearable devices present in the market including some discriminative characteristics.

[3] https://www.microsoft.com/it-it/hololens

[4] https://www.tobiipro.com/product-listing/tobii-pro-glasses-2/

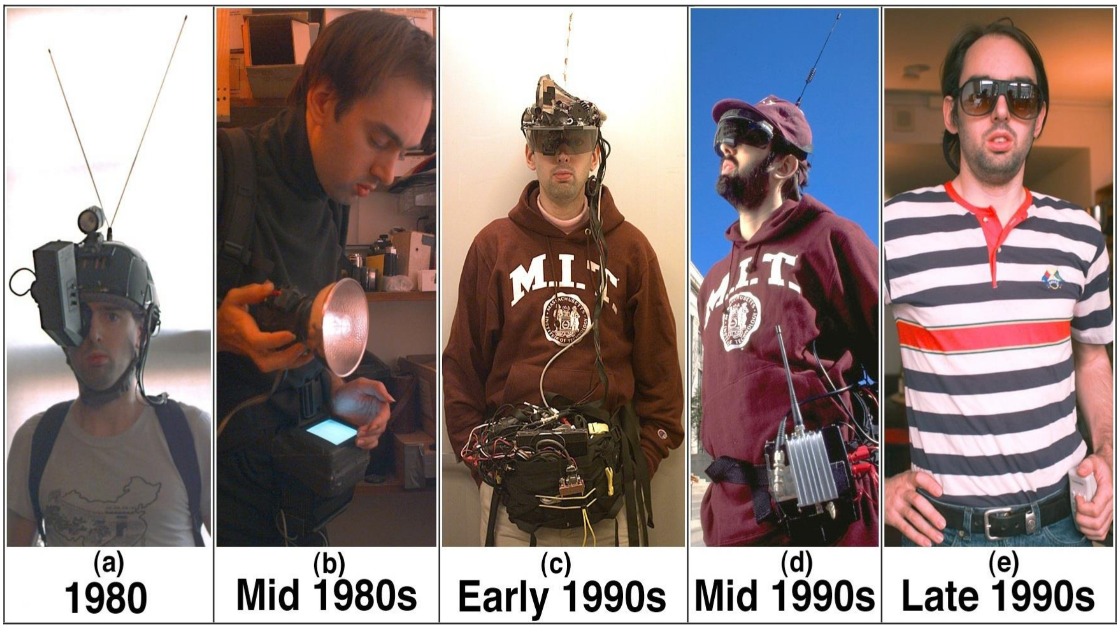


Figure 1.3: The temporal evolution of the wearable computers developed by Mann.

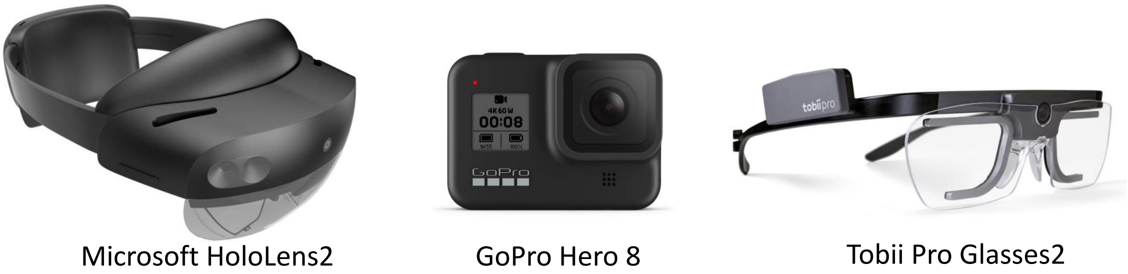


Figure 1.4: Examples of three wearable devices with different characteristics.

1.3 Aim and Approach

The aim of this thesis is to study the human behavior from the first person point of view considering both cultural heritage and industrial domains. We choose the First Person Vision paradigm because we think that data acquired with these systems contain information useful to understand the surrounding environment and the intentions of a human.

To build a first person vision system able to understand the cultural heritage scenario, we acquired two egocentric datasets: UNICT-VEDI 3.2 and Egocentric Cultural Heritage (EGO-CH) 3.3. These datasets represent the first egocentric

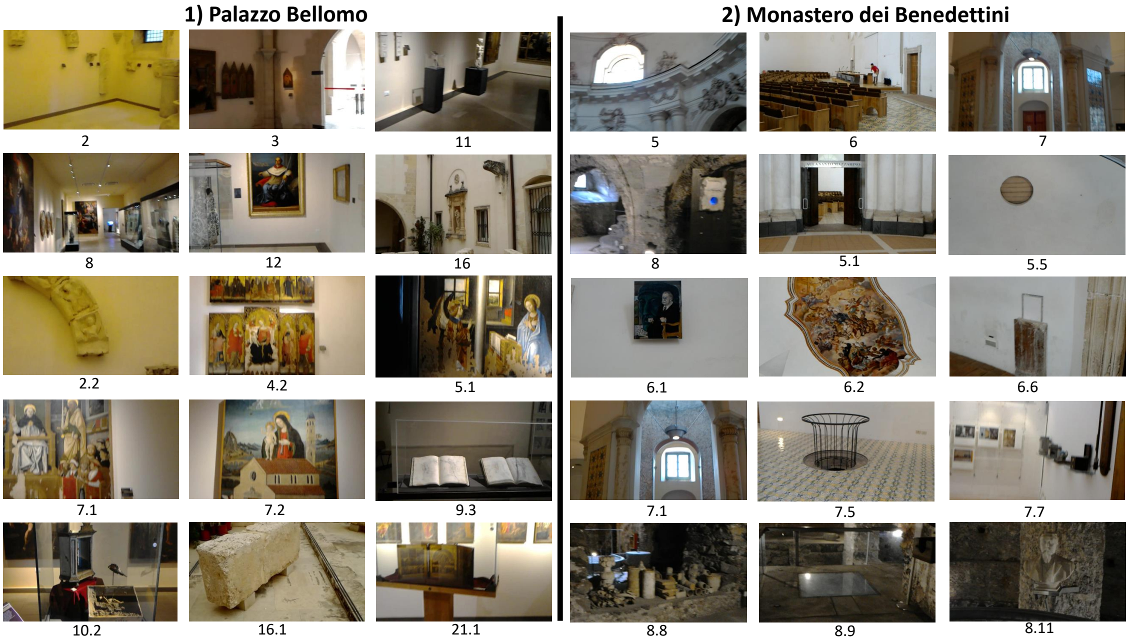


Figure 1.5: Examples of first person vision data acquired in the two cultural sites.

datasets acquired in this domain. The datasets are publicly available to encourage researchers to study the human behavior in this domain. To explore additional egocentric problems related to the cultural sites, we extend these datasets providing additional data and annotations. We consider two different cultural sites in Sicily to explore the generalization aspects of the proposed egocentric system. Figure 1.5 shows some frames acquired in the considered cultural sites.

The first service that a complete FPV system has to take into account is the localization of the visitors. We tackle the room-based localization problem considering egocentric videos acquired by real visitors. We address the problem considering a supervised approach which performs temporal segmentation of the locations where the visitor has been. With a temporal segmentation of the visitors, it is possible to create automatically a video-summary of the visit to give to the visitor at the end of the visit.

The natural second step is to detect and recognize the objects observed by the visitors. A system which can perform the detection of the observed objects can provide additional information regarding the object using the augmented reality. Firstly, we define what is a point of interest and explore the difference with respect to a general object, then, we benchmark a temporal segmentation approach and a standard object detector. To support our analysis on the Point of Interest task, we annotated the UNICT-VEDI dataset with bounding boxes around the points of interest.

We further explore the semantic object segmentation to obtain more information regarding the observed objects, i.e., the shape of the object. Considering the effort needed to annotate the data with semantic masks, we exploited synthetic data and image-to-image translation approach to reduce the domain shift gap between the real and the synthetic domains.

We address the problem of object retrieval which consists in retrieving an image of the same observed object from a database. This task can be useful to perform automatic recognition of artworks when detection can be bypassed, i.e., when the user places the artwork in the center of the field of view using a wearable or mobile device. We obtain a set of query images by extracting image patches from the bounding boxes annotated in the EGO-CH dataset. We consider two variants of this task. In the first variant, object retrieval is framed as a one-shot retrieval problem. In the second variant, we split the set of image patches into a training set (which represents a DB) and a test set.

Moreover, we also try to understand if given an egocentric video acquired by a real visitor is it possible to understand which point of interest has been remembered at the end of the visit and how much did he like them? To explore this proposed problem, we submitted to the visitors surveys at the end of each visit to use as ground truth to evaluate our approach. We address the problem considering a baseline which takes as input the temporal annotations indicating the objects observed by the visitors.

All these tasks have allowed the design and development of a complete egocentric system which is able to assist the visitors during their visits. The proposed system is called VEDI. The wearable system also provides useful information for the cultural manager. The VEDI system is shown in Figure 1.6.

All the provided services are the results of the studies performed on the aforementioned problems in cultural sites.

We further explore the human behavior also considering the industrial domain. We define the Egocentric Human-Object Interaction (EHOI) detection as the task

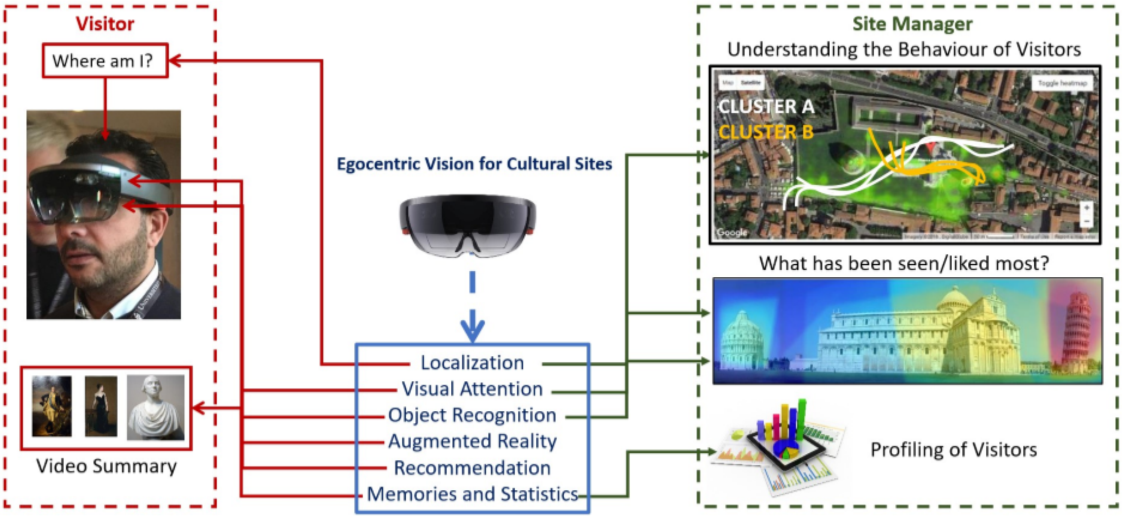


Figure 1.6: Overview of the VEDI system able to assist visitors in cultural sites.

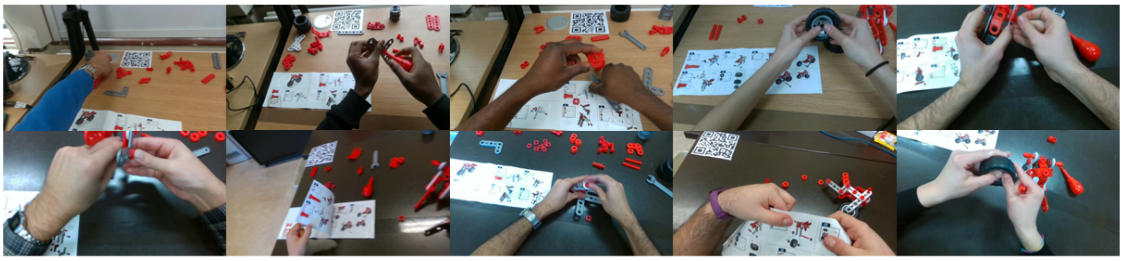


Figure 1.7: Examples of frames belonging to the MECCANO dataset.

of producing (verb, objects) pairs describing the interaction observed from the egocentric point of view. Since there were not public datasets in the literature, suitable to study the human behavior in the industrial domain, due to the fact that data acquisition in industrial domains is difficult because of privacy issues and the need to protect industrial secrets, we acquire and publicly release the MECCANO dataset. MECCANO is the first dataset of egocentric videos acquired in the industrial domain and explicitly annotated to study the human-objects interactions. Figure 1.7 reports some frames belonging to the MECCANO dataset.

We report a complete benchmark on this dataset addressing some fundamental task useful to understand the human behavior in this challenging domain: 1) Action recognition, 2) Active Object Detection, 3) Active Object Recognition, 4) Egocentric

Human-Object Interaction detection.

We also tackle the problem of anticipating next-active objects which could allow to prevent industrial accidents or guarantee workers safety. We further extended MECCANO presenting the MECCANO Multimodal dataset, which includes depth maps, gaze signals and another RGB stream with a wide field of view. We also annotated the dataset to address the next-active object task, with bounding boxes for the hands and the next-active objects. We report some preliminary experiments, considering some frame-based approaches and a video-based approach to consider the temporal relation between frames.

1.4 Relevance of the Considered Tasks

First Person Vision algorithms allow to develop applications which can be used in different domains (e.g., to assist workers in the industrial domain or to help humans during their daily activities). Firstly, we analyze the fundamental tasks useful to develop a FPV system capable to assist a human. Then, we discuss how these tasks could be useful in our lives considering the different domains.

1.4.1 Localization

First Person Vision systems allow to build a localization system based on the analysis of images and videos. These systems can support the users when they perform their daily activities capturing useful information about where they spent their time [6]. A manager of a retail store could use the localization information of its customers [7] to understand where they move and how much time they spent inside the store. The localization information could be useful in the industrial domain if an accident occurs to preserve the safety of workers during an evacuation. We distinguish two levels of localization: 1) point-wise localization and 2) room-based localization. Figure 1.8 reports the comparison of the point-wise and room-based localization. The former is represented by the 2D position of the user in the map of the considered environment (left), instead, the latter indicates the room/area in which the user is located (right). In this thesis, we address the problem of room-based localization in the domain of cultural heritage which has never been studied considering the egocentric perspective.

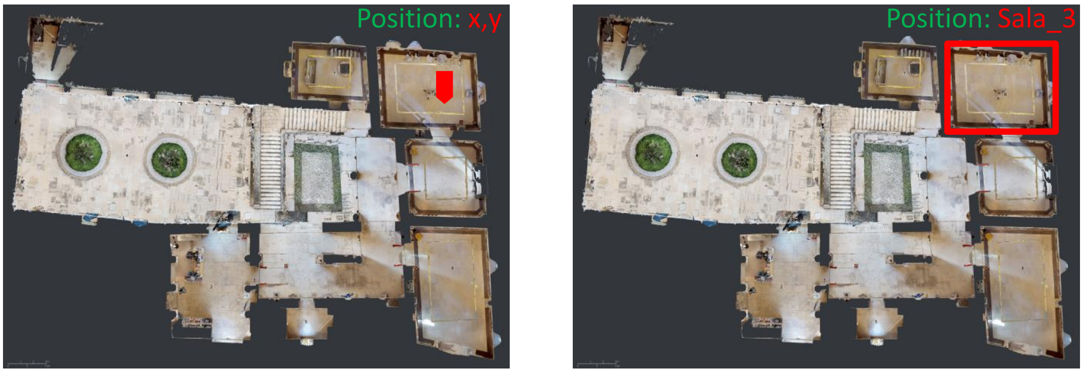


Figure 1.8: Comparison of the point-wise and room-based localization.

1.4.2 Object Detection and Recognition

Detecting and recognizing the objects present in the surrounding environment is a fundamental ability for a first person vision system. This information is useful to understand the context in which the user is, which objects are available, and which actions and interactions the user could perform. Identifying the objects from the first person view is challenging. The appearance of the objects changes frequently due to the variable distance between the user and the objects, the different light conditions, the occlusions, the head-motion which can introduce motion-blur. Moreover, the systems which recognize the objects have to take into account the intra-class variation of objects, they should be able to generalize the object regardless of the different appearance (e.g., shapes or colours). Figure 1.9 shows some examples of intra-class variation related to the "chair" and "wardrobe" objects.

The aim of an object detector is also to localize the object in the scene. Usually, the localization of the object is represented by a bounding box drawn around the considered object. Each bounding box is represented by (x,y,w,h) tuple which indicates its 2D position and its shape. Localizing the objects in the scene is a key information to understand where the objects are with respect to the agent and on which objects the attention of the user is focused, e.g., the user tends to place the object of interest at the center of the frame.

An example of the output of an object detection system is shown Figure 1.10.

In this thesis, we treat the problem of recognizing objects in the cultural heritage and industrial domain from egocentric data.

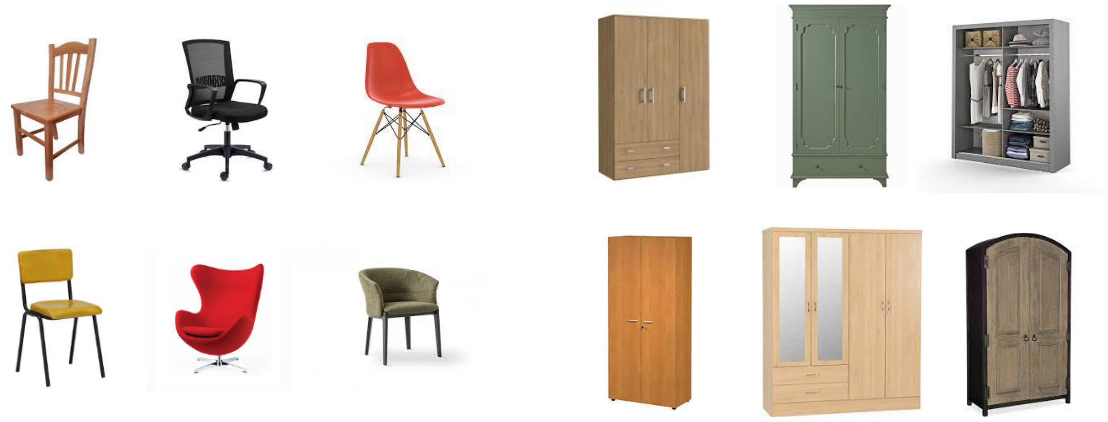


Figure 1.9: Examples of the intra-class variation considering the "chair" (left) and the "wardrobe" (right).

1.4.3 Human-Object Interaction Detection

Understanding the surrounding world is not limited only to the recognition of the objects but also requires to understand how the agent interacts with them. A first person vision system should be able to recognize which objects the human is interacting with. This information can be useful for monitoring the user which performs sequential activities, i.e., a system could alert the agent which forgets to perform a specific step in a considered pipeline of tasks.

Human-Object Interaction (HOI) detection has been generally studied in the context of third person vision [8, 9, 10, 11, 12, 13, 14]. Since we believe that modelling actions both semantically and spatially is fundamental for egocentric vision applications, in this thesis, we instantiate the Human-Object Interaction detection task in the context of the first person vision.

HOI detection consists in detecting the occurrence of human-object interactions, localizing both the humans taking part in the action and the interacted objects. HOI detection also aims to understand the relationships between humans and objects, which is usually described with a verb. Possible examples of HOIs are *"talk on the cell phone"* or *"hold a fresbee"*. HOI detection models mostly consider one single object involved in the interaction [8, 15, 9, 16, 11]. Hence, an interaction is defined as a triplet in the form <*human, verb, object*>, where the human is the subject of the action specified by a verb and an object. Sample images related to human-object interactions in a third-person scenario are shown in Figure 1.11-top.

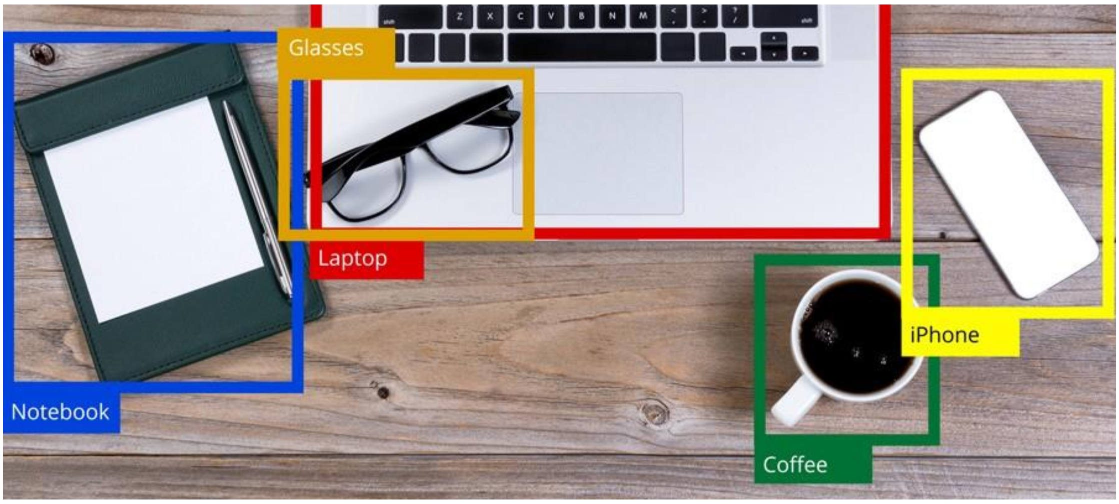
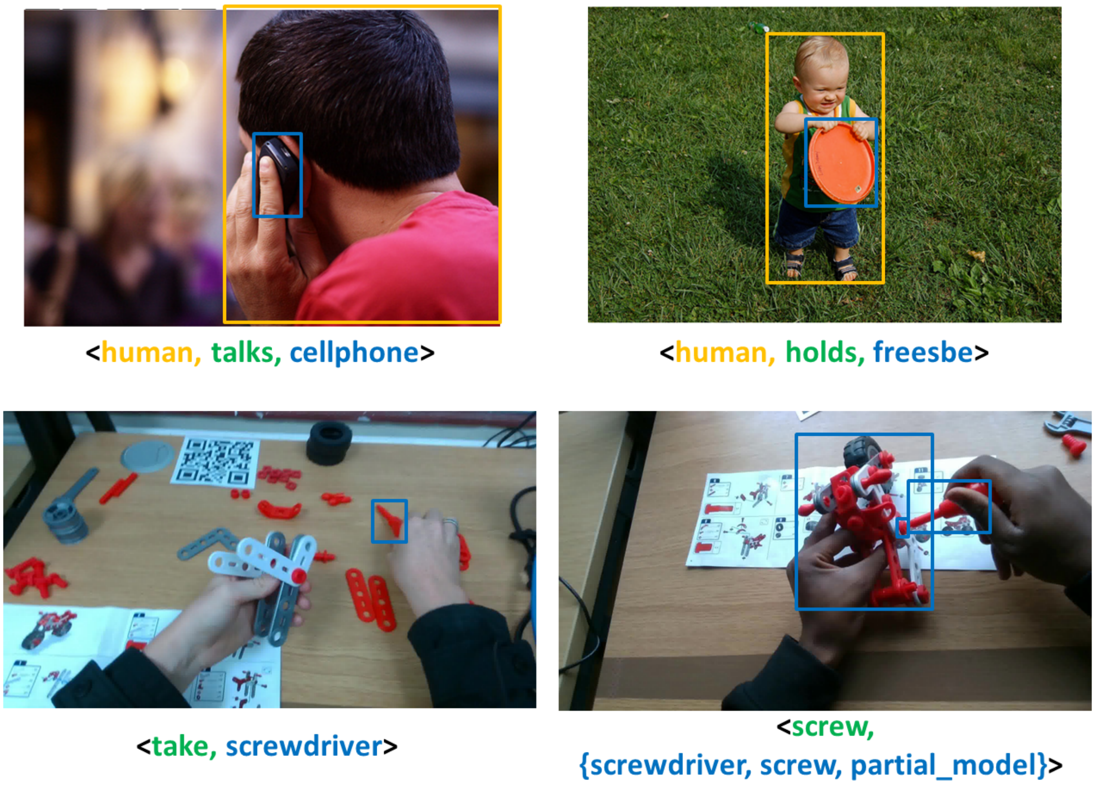


Figure 1.10: Examples of an image processed by a system which detects and recognizes the objects in the scene.

We define Egocentric Human-Object Interaction (EHOI) detection as the task of producing <verb, objects> pairs describing the interaction observed from the egocentric point of view. Note that in EHOI, the human interacting with the objects is always the camera wearer, while one or more objects can be involved simultaneously in the interaction. The goal of EHOI detection is to infer the verb and noun classes, and to localize each active object involved in the interaction. Figure 1.11-bottom reports some examples of Egocentric Human-Object Interactions.

1.4.4 Predicting What Will Happen

Anticipating the human behavior is a challenging problem for which the research community shows a growing interest. The concept of anticipation covers many problems beyond the ones just discussed, i.e., anticipation of the HOI, anticipation of the next position of the user, anticipation of the next action or the next-active objects. A system which anticipates if a human will perform a dangerous interaction could alert him to guarantee his safety. Moreover, in the domain of cultural heritage, an intelligent system will predict which artworks will be most watched or in which areas of the museum the user will go. In our daily lives, a system capable of anticipating the next objects a user will interact with, could assist a person with mental disabilities to remind how to use a specific object.



<human, talks, cellphone>

<human, holds, freesbe>

<take, screwdriver>

<screw, {screwdriver, screw, partial_model}>

Figure 1.11: Examples of Human-Object Interactions in third person vision (first row) and first person vision (second row)[5]

We distinguish three levels of anticipation:

- *short-term*: where the near future is considered;
- *middle-term*: the anticipation is performed just a few seconds before;
- *long-term*: the time horizon of minutes/hours is considered.

Short-term anticipation considers the current state of the scene and predicts what will happen in the near future (i.e., milliseconds). An example is represented by a system which predicts which object will be active considering all the objects present in the environment (see Figure 1.12).

Middle-term anticipation takes into account only a limited knowledge of the current scene. Action anticipation is an example of this level of anticipation where the system should predict one of the possible future actions that can occur. Figure 1.13

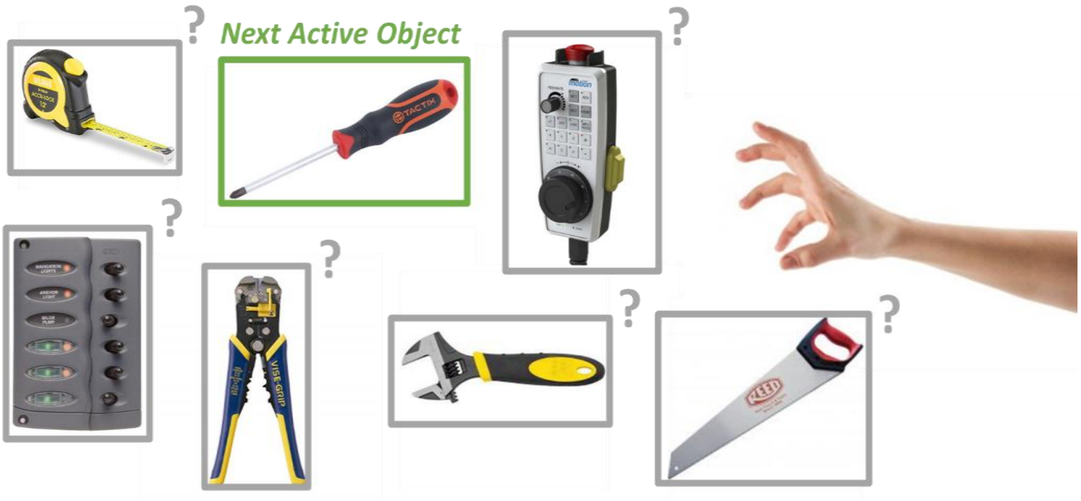


Figure 1.12: Example of next-active object prediction.

reports an example of middle-term action anticipation. Given a past video-shot, the system predicts what will happen after 1 second.

Also, the next-active object prediction task belongs to this case, considering a system which predicts the objects which will be active in one or more seconds. In this case, the target object could not be present in the observed scene considering the dynamic variation of the first person view. This level of anticipation is more complex with respect to the short-term anticipation, due to the indeterminacy and the ambiguity of the future.

Long-term anticipation aims to predict a workflow after observing the past. This level of anticipation could be applied in specific domains where the actions to perform are well defined (e.g., in the kitchens to follow a food recipe).

In this thesis, we address the problem of predicting the next-active objects considering both short and middle levels of anticipation. To this aim we consider the industrial domain with the proposed MECCANO Multimodal egocentric dataset (Section 4.5).

1.5 Contributions

The main contributions of this thesis are the following:

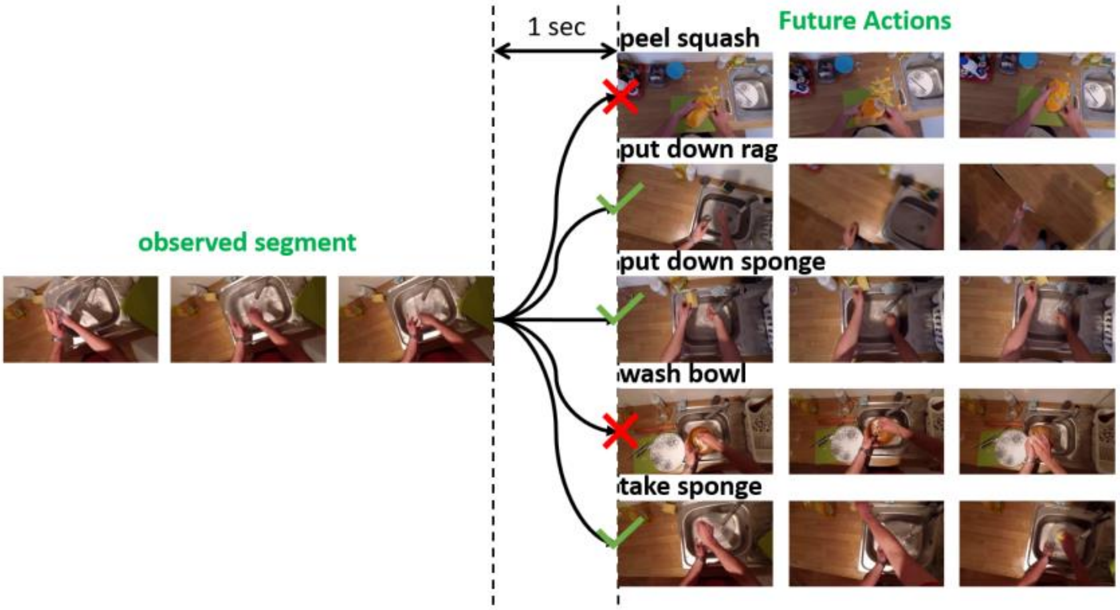


Figure 1.13: Middle-term action anticipation. Given an observed video-shot (left), the system predicts the next action 1 second before that it happens.

- The introduction of two labeled egocentric datasets (UNICT-VEDI and Egocentric Cultural Heritage (EGO-CH)) acquired in the domain of cultural heritage aimed to understand the human behavior in two different cultural sites;
- The definition of the "Point of Interest" (POI) concept in the domain of cultural heritage with respect to the classic object definition;
- A detailed benchmark addressing fundamental tasks to study human behavior in cultural sites: room-based localization, points of interest recognition, object retrieval, survey generation and semantic object segmentation;
- The presentation of the VEDI system which is able to assist both visitors and cultural managers in the domain of cultural heritage;
- The definition of the Egocentric Human-Object Interaction detection task from first person videos;
- The acquisition and annotation of the MECCANO dataset, which is the first dataset related to the industrial-like domain;

- A benchmark on the MECCANO dataset of different state-of-the-art methods on four proposed tasks useful to understand the human behavior in the industrial domain: action recognition, active object detection, active object recognition and egocentric human-object interaction (EHOI) detection;
- The introduction of the MECCANO Multimodal dataset which comprises depth maps, gaze signals and a second RGB stream with a wide field of view with respect to the previous version (MECCANO);
- The formulation of the next-active objects problem from the egocentric point of view;
- The preliminary experiments on the MECCANO Multimodal dataset aimed to anticipate the next-active objects.

The scientific contribution of this thesis have been disseminated with a patent and publications in international journals and conferences:
Patents:

- G. M. Farinella, A. Furnari, F. Ragusa, E. Ragusa, G. Sorbello, A. Lopes, L. Santo, M. Samarotto, B. Scarso, E. Scarso, ”Metodo di assistenza virtuale relativo dispositivo e sistema”, Università degli Studi di Catania, Xenia Progetti s.r.l., Morpheos s.r.l., Patent Application number: 102020000027759, 19/11/2020

International Journals:

- F. Ragusa, A. Furnari, S. Battiato, G. Signorello, G. M. Farinella. EGO-CH: Dataset and Fundamental Tasks for Visitors Behavioral Understanding using Egocentric Vision. Pattern Recognition Letters - Special Issue on Pattern Recognition and Artificial Intelligence Techniques for Cultural Heritage (PRL), 2020
- F. Ragusa, A. Furnari, S. Battiato, G. Signorello, G. M. Farinella. Egocentric Visitors Localization in Cultural Sites. In Journal on Computing and Cultural Heritage (JOCCH), 2019

International Conferences:

- F. Ragusa, A. Furnari, S. Livatino, G. M. Farinella. The MECCANO Dataset: Understanding Human-Object Interactions from Egocentric Videos in an Industrial-like Domain. IEEE Winter Conference on Applications of Computer Vision (WACV), 2021 (ORAL)

- F. Ragusa, D. DiMauro, A. Palermo, A. Furnari, G. M. Farinella. Semantic Object Segmentation in Cultural Sites using Real and Synthetic Data. International Conference on Pattern Recognition (ICPR), 2020

- G. M. Farinella, G. Signorello, S. Battiato, A. Furnari, F. Ragusa, R. Leonardi, E. Ragusa, E. Scuderi, A. Lopes, L. Santo, M. Samarotto. VEDI: Vision Exploitation for Data Interpretation. In 20th International Conference on Image Analysis and Processing (ICIAP), 2019

- F. Ragusa, A. Furnari, S. Battiato, G. Signorello, G. M. Farinella. Egocentric Point of Interest Recognition in Cultural Sites. In 14th International Conference on Computer Vision Theory and Applications (VISAPP), 2019

- F. Ragusa, L. Guarnera, A. Furnari, S. Battiato, G. Signorello, G. M. Farinella. Localization of Visitors for Cultural Sites Management. In International Conference on Signal Processing and Multimedia Applications (SIGMAP), Porto, Portugal, July 26-28, 2018 (ORAL)

1.6 Outline

The thesis is divided into 7 chapters.
Chapter 2 reports a deep analysis of the state-of-the-art considering all the problems treated in this thesis.
Chapter 3 investigates the visitors behavior understanding in cultural sites considering 5 fundamental tasks.
Chapter 4 defines the task of Egocentric Human-Object Interaction (EHOI) detection and investigates the human behavior in an industrial domain. Moreover, it analyzes the difference between the concepts of *action* and *interaction* and defines the next-active objects detection task.
Chapter 5 summarized the obtained findings and analyzes the current limitations

of the developed technologies. Finally, it also concludes the thesis and gives insights for future directions.

Chapter 2

Related Work

This chapter, analyzes the literature in details, treating the topics at the core of the works presented in this thesis. In Section 2.1, we describe past works related to the intelligent vision systems developed to assist the human in the cultural heritage domain. Firstly, we discuss the first person vision datasets present in the literature in Section 2.1.1. The Room-based Localization and Object Detection and Recognition problems are treated in Section 2.1.2 and Section 2.1.3 respectively. The main studies of the literature regarding the Human-Object Interaction problem in industrial domain, are analyzed in Section 2.2. The first person vision datasets present in the literature regarding the industrial domain are treated in Section 2.2.3. Finally, the works related to the Next-Active Objects detection problem are discussed in Section 2.2.4.

2.1 Visitors Behavioural Understanding in Cultural Sites

The use of Computer Vision to improve the fruition of cultural objects has been already investigated in past studies. Cucchiara and Del Bimbo discuss the use of computer vision and wearable devices for augmented cultural experiences in [17]. In [18] it is presented the design for a system to provide context aware applications and assist tourists. In [19] it is described a system to perform real-time object classification and artwork recognition on a wearable device. The system makes use of a Convolutional Neural Network (CNN) to perform object classification and artwork classification. In [20] is discussed an approach for egocentric image classification and object detection based on Fully Convolutional Networks (FCN). The system is adapted to mobile devices to implement an augmented audio-guide. In [21], it is proposed a method to exploit georeferenced images publicly available on social media platforms to get insights on the behavior of tourists. In [22] is addressed the problem of creating a smart audio guide that adapts to the actions and interests of visitors. The system uses CNNs to perform object classification and localization and is deployed on a NVIDIA Jetson TK1. In [23] is investigated multimodal navigation of multimedia contents for the fruition of protected natural areas. Razavian et al. [24] proposed a method to estimate the attention of the visitors of an exhibition, whereas Raptis et al. [25] studied the design of mobile applications in museum environments and highlighted that context influences interaction.

Past works investigated specific applications, generally relying on data collected on purpose and not publicly released. In the works presented in Chapter 3 of this thesis, we aim to standardize the fundamental problems of visitors behavioral understanding in cultural sites by proposing public datasets and a series of tasks. Morever, differently from the aforementioned works, the proposed VEDI system (see Section 3.9) has been designed to both support the visitors of cultural sites and provide useful behavioral information to the site manager.

2.1.1 First Person Vision Datasets

Previous works have proposed third person vision datasets to tackle many challenging tasks in the cultural heritage domain (e.g. localization, artworkd detection, semantic segmentation, etc.). In this section, we highlight the lack of egocentric datasets in the literature analyzing past works which considered the cultural heritage and industrial domains. Few image-based datasets have been proposed in past works to investigate different problems related to cultural sites. For instance, in [20, 26], it is proposed a dataset of images acquired by using third person or head-mounted cameras. The dataset contains a small number of images and is intended to address the problem of image search (e.g., recognizing a painting). In [27], it is presented a dataset acquired inside the National Museum of Bargello in Florence. The dataset (acquired using 3 fixed IPcameras) is intended for people and group detection, gaze estimation and behavior understanding. Koniusz et al. [28] proposed the OpenMIC dataset containing photos captured in ten different exhibition spaces of museums to explore the problem of artwork identification. DelChiaro et al. [29]

proposed NoisyArt, a dataset composed of artwork images collected from Google Images and Flickr correlated by metadata gathered from DBpedia.

Different datasets such as Pascal VOC [30] and COCO [31] have been proposed to explore the problem of semantic object segmentation. Despite these datasets are useful benchmarks to design algorithms, when objects have to be recognized at the instance level, as it is the case of cultural sites, it is necessary to fine-tune such algorithms with domain-specific data. Hence, it is required to collect and manually label images of the specific objects of interest. This procedure is generally laborious and expensive. The availability of synthetic data would in principle enable to train semantic segmentation models at a lower cost. Synthetic datasets have been proposed in the past considering virtual 3D environments to generate semantic labels in a simple way [32, 33]. Some works have considered photo-realistic images obtained through augmented reality [34], whereas others have generated synthetic clones from a small amount of real data [35]. However, such datasets do not generally include both real and synthetic images of the same object annotated with semantic masks at the instance level.

In contrast with the aforementioned works where data are related third person, in this thesis we described the proposed datasets composed of egocentric videos, and release it publicly. The datasets can be used to address different tasks related to visitors behavioral understanding in cultural sites. A significant part of the proposed datasets has been collected by real visitors (i.e., 60 visitors) to create a realistic set of data for benchmarking. The proposed datasets are:

- UNICT-VEDI (Section 3.2);
- UNICT-VEDI-POIs (Section 3.2.3);
- Egocentric Cultural Heritage (EGO-CH) (Section 3.3);
- EGO-CH-OBJ-SEG (Section 3.3.3);

2.1.2 Room-Based Localization

Person localization can be tackled outdoor using Global Positioning System (GPS) devices. These systems, however, are not suitable to localize the user in an indoor environment. Therefore, different Indoor Positioning Systems (IPS) have been proposed through the years [36]. In order to retrieve accurate positions, these systems rely on devices such as active badges [37] and WiFi networks [38], which need to be placed in the environments and hence become part of the infrastructure. This operational setting is not scalable since it requires the installation of specific devices, which is expensive and not always feasible, for instance, in the context of cultural heritage but also in the industrial domain.

Visual localization can be used to overcome many problems. For instance, previous works addressed visual landmark recognition with smartphones [39, 40, 41]. In particular, the use of a wearable cameras allows to localize the user without relying on specific hardware installed in the considered site. Visual localization can be performed at different levels, according to the required localization precision and to the amount of available training data. Three common levels of localization are scene recognition [42, 43], location recognition [44, 45, 46, 6] and 6-DOF camera pose estimation [47, 48, 49]. Some works also investigated the combination of classic localization based on non-visual sensors (such as bluetooth) with computer vision [50, 51].

In the works presented in this thesis, we concentrate on location recognition in the domain of cultural sites, since we want to be able to recognize the environment (e.g., room) in which the visitor is located. Location recognition is the ability to recognize when the user is operating in a specific space at the instance level. In this case the egocentric (first person) vision system should be able to understand if the user is in a given location. Such location can either be a room (e.g., office 368 or exhibition room 3) or a personal space (e.g., office desk). In order to setup a location recognition system, it is usually necessary to acquire a moderate amount of visual data covering all the locations visited by the user. Visual location awareness has been investigated by different authors over the years. In [44], it has been addressed the recognition of basic tasks and locations related to the Patrol game from egocentric videos in order to assist the user during the game. The system was able to recognize the room in which the user was operating using simple RGB color features. An Hidden Markov Model (HMM) was employed to enforce the temporal smoothness of location predictions over time. In [45], it was proposed a system to recognize personal locations from egocentric video using the "approaching trajectories" observed by the wearable camera. At training time the system built a dictionary of visual trajectories (i.e., collections of images) captured when approaching each specific location. At test time, the observed trajectories are matched to the dictionary in order to detect the current location. In [46], it has been designed a context-based vision system for place and scene recognition. The system used an holistic visual representation similar to GIST to detect the current location at the instance level and recognize the scene category of previously unseen environment. Other authors [52] proposed a way to provide context-aware assistance for indoor navigation using a wearable system. When it is not possible to acquire data for all the locations which might be visited by the user, it is generally necessary to explicitly consider a rejection option, as proposed in [6]. This thesis focuses on room-based localization task considering two bigger egocentric datasets acquired in the domain of cultural heritage. This task has been addressed in a real scenario where real visitors acquired egocentric videos during their visits in the cultural sites. The room-based localization task has never be addressed in a real egocentric scenario related to the cultural heritage.

2.1.3 Object Detection and Recognition

Different works investigated how to detect and recognize objects to describe an image, localize the objects in the scene to enable a robot to assist a person who suffers from some disorder, and to perform tracking of a specific object. The authors of [53] and [54] proposed deep models for object recognition. Some approaches classify image patches extracted from region proposals [53, 55, 56], whereas others classify a fixed set of evenly spaced square windows [54]. The authors of [57] introduced the ideas of prior box and region proposal network. As an evolution of [55], the authors of [58] replaced the heuristic region proposal with RPN (Region Proposal Network) inspired by MultiBox [57]. The authors of [59] leveraged RPN to directly classify objects inside each prior box. The authors of [60] extended FasterRCNN by adding a branch for predicting class-specific object masks, in parallel with the existing bounding box regressor and object classifier. The third version of YOLO [61], which is considered a state-of-the-art real-time object detector, uses a novel multi-scale training method and, following the authors of [62], proposed a technique to jointly train on object detection and classification. A recent work on optimization methods to train deep networks for object detection and segmentation is reported

in [63]. The approach proposed in [64] detects an object bounding box as a pair of keypoints (top-left corner and bottom-right corner) using a single CNN. An improvement to bounding box localization has been proposed in [65] where IoU-Net is introduced.
To the best of our knowledge object detection and recognition in the context of cultural sites has been less investigated. This is probably due to the absence of large datasets in this context. Seidenari et. al [22] and Taverriti et al. [19] proposed to perform object classification and artwork recognition to assist tourists with additional information about the observed objects. The authors of [28] proposed a new dataset (OpenMIC) that contains photos captured in 10 distrinct exhibition spaces of several museums and explored the problem of artwork identification. In general, object detectors (e.g., YOLOv3 [61]) have been used to detect artworks in cultural sites. However, it should be noted that, as pointed out in Section 3.1, depending on the cultural site, not all Points Of Interest are objects. For instance, a point of interest can be an architectural element such as a pavement, or even a corridor. In this case, it should be considered that object detectors can be limited. In this thesis we consider this last aspect.

Semantic Object Segmentation

The aim of a semantic segmentation algorithm is to assign a class label to each pixel of a given input image. Several approaches to semantic segmentation have been proposed through the years. In particular, recent works train CNNs for pixel-wise classification in a fully supervised fashion. Among the most notable approaches, the authors of [66] proposed fully convolutional networks, which generalize CNNs for image classification to perform semantic segmentation. The authors of [67] introduced SegNet, an encoder-decoder architecture based on VGG [68]. The authors of [69] investigated the use of up-sampled convolutional filters to enlarge receptive fields, spatial pyramid pooling to segment objects at multiple scales, and probabilistic graphic models to improve the localization of object boundaries. The authors of [70] introduced RefineNet to exploit multi-level features in a recursive manner to generate high-resolution semantic feature maps. The authors of [71] designed a pyramid scene parsing network (PSPNet) to spatially enhance pixel-level features using global pyramid pooling. The authors of [72] introduced a Semantic Prediction Guidance (SPG) which learns how to re-weight local features across prediction stages. The authors of [73] presented SceneAdapt, a scene-based domain adaptation approach for semantic segmentation. This thesis propose a dataset for object segmentation and related benchmark in the context of cultural heritage (Section 3.8).

Domain Adaptation

The aim of the domain adaptation task is to reduce the performance drop caused by the distribution misalignment between source and target domain. This task is mostly studied using conventional approaches [74, 75, 76] or methods based on CNNs [77, 78, 79, 80, 81].

Synthetic data transformation to realistic style has been explored inspired by the advent of the generative adversarial networks [82, 83] and image translation approaches [84]. The authors of [85] presented a method to perform image translation from a source domain to a target domain without the presence of paired examples. In Section 3.8 of this thesis, we explore the use of both real and synthetic data to perform semantic object segmentation in the cultural heritage domain, using also an image-to-image translation approach to reduce the gap between the two domains (i.e., real and synthetic).

Object Retrieval

Many previous works investigated approaches to image retrieval. Rubhasy et al. [86] used an ontology-based approach to retrieval in multimedia cultural heritage collections. The goal is to enable the integration of different types of cultural heritage media and to retrieve relevant heritage media given a query. Kwan et al. [87] proposed matrix of visual perspectives to address Content-based Image Retrieval (CBIR) of cultural heritage symbols, whereas Iakovidis et al. [88] perform pattern-based Content-based Image Retrieval. The work of [89] focused on discarding image outliers using Content-based Image Retrieval. Despite the availability of advanced approaches, for generality and ease of comparison, in this thesis we consider simple baselines based on image representation and nearest neighbor search to address the object retrieval task in the domain of cultural sites (Section 3.6).

2.2 Human-Object Interaction Detection in Industrial Scenarios

HOI detection from third person images is an active area of reaserch. The work of [8] was the first to explore the HOI detection task annotating the COCO dataset [31] with verbs. The authors of [8] proposed a method to detect people performing actions able to localize the objects involved in the interactions on still images. The authors of [9] proposed a human-centric approach based on a three-branch architecture (InteractNet) instantiated according to the definition of HOI in terms of a <human, verb, object> triplet. This approach analyzes each human-object pairs detected with an object detector [58] using a heat map to represent their relationship. Some works [10, 11, 12] explored HOI detection using graph convolutional neural networks after detecting humans and objects in the scene. Recent works [13, 14] represented the relationship between both, the humans and the objects, as the intermediate point which connects the center of the human and object bounding boxes.

The aforementioned works addressed the problem of HOI detection in the third person vision domain, whereas this thesis focuses on the task of HOI detection from an egocentric perspective considering the proposed MECCANO dataset (Section 4.4.

2.2.1 Action Recognition

Video action recognition has been thoroughly studied by researchers, especially from the third person view. Some works [90, 91, 92, 93, 94] mixed classic approaches considering hand-crafted features, such as optical flow, and deep networks to represent the motion of actions using two stream networks. 3D ConvNets are commonly used to encode both spatial and temporal dimensions in a unified way [95, 96, 97]. Long-term filtering and pooling has focused on representing actions considering their full temporal extent [98, 93, 94, 99]. Other works separately control spatial and temporal dimensions factoring convolutions into separate 2D spatial and 1D temporal filters [100, 101, 102, 103]. Slow-Fast networks [104] avoid using pre-computed optical flow and encodes the motion of actions into a “fast” pathway (which operates at a high frame rate) and simultaneously a “slow” pathway which captures semantics (operating at a low frame rate). The authors of [99] introduced a network module called Temporal Relation Network (TRN) to learn temporal relations between video frames at multiple time scales. In [105], a temporal shift module (TSM) has been proposed. This module allows 2D architecture to obtain comparable performance to 3D CNNs. Previous works also investigated egocentric action recognition adapting third person vision approaches to the first person scenario [105, 99, 104, 106]. In this thesis, we asses the performance of state-of-the-art action recognition methods on the proposed MECCANO dataset considering SlowFast network [104] as a baseline (Section 4.4).

2.2.2 Egocentric Human-Object Interaction Detection

The problem of Egocentric Human-Object Interaction (EHOI) detection is understudied due to the limited availability of egocentric datasets labelled for this task. Some studies have modeled the relations between entities for interaction recognition as object affordances [107, 108, 109]. Other studies tackled tasks related to EHOI recognition proposing hand-centric methods [110, 111, 112]. The authors of [112] proposed to detect and localize hands in the scene distinguishing left from right hands. Objects are classified into two classes: *active* or *passive*. In particular, if the hands are in contact with one or more objects, the object is considered as *active*, otherwise, it is considered as *passive*. The authors of [113] proposed to search network structures with differentiable architecture to construct adaptive structures for different videos to facilitate adaptive interaction modeling. The method has been evaluated on the Something-Something dataset [114] which contains egocentric-like videos.

Despite these related works have considered human-object interaction from an egocentric point of view, the EHOI detection task has not yet been studied systematically. In this thesis we define the task of EHOI detection and we focus on this problem related to the industrial domain, considering the proposed MECCANO dataset (Section 4.4).

2.2.3 First Person Vision Datasets

A few datasets have been proposed for the task of Human-Object Interaction (HOI) detection. These datasets are generally composed of static images [8, 115] or videos

[116, 13]. In particular, the authors of [8] annotated the COCO dataset [31] with verbs (V-COCO) to study the problem of detecting HOI. V-COCO includes 10346 images annotated with 26 actions. HICO-Det [115] is a large-scale dataset composed of static images used as a benchmark to study the task of HOI detection. This dataset includes 47766 images and has been annotated with 117 verbs and 80 objects (same objects of COCO). While these datasets focused on common and general actions, the HOI-A dataset [13] focused on a subset of actions, such as *smoking cigarette* or *talk on mobile phone* which can be considered dangerous actions while driving. The dataset is composed of 38668 images annotated with 10 verbs and 11 object classes. Many efforts have been devoted to the collection, labeling, and release of action recognition datasets. Among these, ActivityNet [116] is a large-scale dataset composed of videos depicting activities that are related to how humans spend their time in their daily lives such as *walking the dog* or *hand-washing clothes*. The dataset is composed of a total of 849 video hours including 203 activity classes. The authors of [117, 118] presented Kinetics, which is a third person video dataset related to human actions. The dataset is composed of 700 human action classes which include human-object interactions such as *play instrument* and human-human interactions such as *shake hands*. For each action there are at least 600 video clips taken from YouTube videos. The authors of [114] proposed Something-Something, a video dataset which includes low-level concepts ("*something-something*") to represent simple everyday aspects of the world. It contains 108499 short videos (from 2 to 6 seconds) annotated with 174 textual description such as "turning *something* upside down" or "spilling *something* next to *something*".

Other works have considered the egocentric scenario. Among these, EPIC-Kitchens and its extension [106, 119, 120] constitute a series of egocentric datasets focused on unscripted activites related to kitchens. In particular, EPIC-Kitchens-55 [106] is composed of 432 videos annotated with 352 classes of objects and 125 different verbs. EPIC-Kitchens-100 [120] is an extension of EPIC-Kitchens-55 [106] in terms of videos (700), environments (45) and hours (100). Along with the dataset, the authors of [120] proposed 6 challenges to study human behavior in the kitchen: action recognition, action detection, action anticipation, domain adapatation for action recognition, object detection and multi-instance retrieval. The authors of

Dataset	Settings	EGO?	Video?	Tasks	Year	Frames	Sequences	AVG. video duration	Action classes	Object classes	Object BBs	Participants
MECCANO	Industrial-like	✓	✓	EHOI, AR, AOD, AOR	2020	299,376	20	20.79 min	61	20	64,349	20
EPIC-KITCHENS [106]	Kitchens	✓	✓	AR, AOR	2018	11,5M	432	7.64 min	125	352	454,255	32
EGTEA Gaze+ [121]	Kitchens	✓	✓	AR	2018	2,4M	86	0.05 min	106	0	0	32
THU-READ [124]	Daily activties	✓	✓	AR	2017	343,626	1920	7.44 min	40	0	0	8
ADL [123]	Daily activities	✓	✓	AR, AOR	2012	1,0M	20	30.0 min	32	42	137,780	20
CMU [122]	Kitchens	✓	✓	AR	2009	200,000	16	15.0 min	31	0	0	16
Something-Something [114]	General	X	✓	AR, HOI	2017	5,2 M	108,499	0.07 min	174	N/A	318,572	N/A
Kinetics [118]	General	X	✓	AR	2017	N/A	455,000	0.17 min	700	0	0	N/A
ActivityNet [116]	Daily activites	X	✓	AR	2015	91,6 M	19,994	2.55 min	200	N/A	N/A	N/A
HOI-A [13]	General	X	X	HOI, AOR	2020	38,668	N/A	N/A	10	11	60,438	N/A
HICO-DET [115]	General	X	X	HOI, AOR	2018	47,776	N/A	N/A	117	80	256,672	N/A
V-COCO [8]	General	X	X	HOI, OD	2015	10,346	N/A	N/A	26	80	N/A	N/A

Table 2.1: Comparative overview of relevant datasets. HOI: HOI Detection. EHOI: EHOI Detection. AR: Action Recognition. AOD: Active Object Detection. AOR: Active Object Recognition. OD: Object Detection.

[121] studied egocentric video action recognition to determine what a person is doing (action recognition) and where they are looking at (gaze estimation). They presented the dataset EGTEA Gaze+ [121], where 32 subjects perform 7 different meal preparation tasks in different kitchens. EGTEA Gaze+ is composed of 106 action classes and includes gaze information acquired at every frame. The authors of [122] released the CMU Multi-Modal Activity Database (CMU-MMAC) to study human activities in a kitchen environment. The authors built a kitchen and acquired egocentric videos from 5 different subjects cooking 5 recipes. They captured RGB videos using different cameras, audio and motion capture information. Egocentric activities related to daily living have been studied in [123], where the ADL dataset has been released. The dataset is composed of one million frames acquired by 20 people performing a set of 18 actions of daily activities in their own apartment. The egocentric action recognition task has been explored considering also the 3D structure of the scenes [124]. The authors of [124] proposed a video-based RGB-D egocentric dataset (THU-READ) including different types of daily-life actions. The egocentric videos have been captured in 5 scenarios such as laboratory, bathroom, conference room, dormitory and restaurant by 8 different subjects performing 40 different actions.

Table 2.1 compares the aforementioned datasets with respect to the proposed MECCANO dataset (see Section 4.3). MECCANO is the first dataset of egocentric videos collected in an industrial-like domain and annotated to perform Egocentric Human-Object Interaction (EHOI) Detection. It is worth noting that previous egocentric datasets have considered scenarios related to kitchens, offices, and daily-life activities and that they have generally tackled the action recognition task rather than EHOI detection.

Multimodal Egocentric Datasets

Different third person datasets [125, 126, 127, 128, 129, 130, 131, 132] have been proposed to study the human behavior using also the depth signal, especially after the release of the Microsoft Kinect [133].

A few works have considered the egocentric scenario. THU-READ [124] and EGTEA-GAZE+ [121] (which have been already discussed in previous section) are the most popular multimodal egocentric datasets used to study the human behavior from the first person point of view considering the depth and gaze signals. The authors of [7] proposed the EgoCart dataset to address the problem of image-based indoor localization in retail stores. The dataset, which is composed of 9 videos, has been acquired using two ZED stereo cameras[1] which have been placed on a shopping cart. The whole dataset is composed of 19531 RGB frames with the associated depth maps. The problem of 3D hand-object actions recognition has been addressed by the authors of [134], which released the Daily hand-object actions dataset containing 1175 videos belonging to 45 action categories. The dataset has been acquired by 6 actors over 3 different scenarios. A total of 105.459 RGB-D frames have been acquired and annotated with hand pose and action categories. The egocentric action recognition problem has been also addressed in [135]. The dataset proposed in [135] is composed of 5 sequences acquired using an RGB-D camera by 4 different users. Not only depth and gaze signals have been considered in past works. Sensor data like accelerometer or gyroscope have been considered to recognize egocentric activity [136]. The dataset was captured using a Google Glass that acquired RGB videos and sensor information. In particular, 200 short sequences have been acquired by 20 different subjects which performed daily activities. The authors of [137] focused on object manipulation and proposed the Grasp UNderstanding (GUN-71) dataset which is composed of 12.000 RGB-D images labeled over 71 grasp classes. The videos have been acquired by 8 different subjects which performed different grasps on personal objects in 5 different houses. The camera used is a chest-mounted Intel'Senz3D[2] which is a webcam with a depth sensor.

Beyond the aforementioned datasets which considered only one extra modality except the RGB signal, the authors of [138] proposed the Gaze-in-Wild dataset

[1] https://www.stereolabs.com/zed/

[2] https://it.creative.com/p/archived-products/blasterx-senz3d

Dataset	Settings	EGO?	Video?	Multimodality	Tasks	Year	Frames	Sequences	AVG. video duration	Action classes	Object classes	Object BBs	Participants
MECCANO Multimodal	Industrial-like	✓	✓	2 RGB, depth, gaze	EHOI, AR, AOD, AOR, NAO	2021	299,376	20	20.79 min	61	20	307,601	20
EgoCart [7]	Retails	✓	✓	RGB, depth	LOC	2021	19,531	9	N/A	0	0	0	N/A
Gaze-in-wild [138]	Daily activities	✓	✓	RGB, depth, gaze	AR	2020	N/A	N/A	N/A	4	0	0	19
EGTEA Gaze+ [121]	Kitchens	✓	✓	RGB, gaze	AR	2018	2,4M	86	0.05 min	106	0	0	32
Daily Hand-Object Actions [134]	Daily activities	✓	✓	RGB, depth	AR, HPE	2018	105,459	1175	0.05 min	45	26	N/A	6
THU-READ [124]	Daily activties	✓	✓	RGB, depth	AR	2017	343,626	1920	7.44 min	40	0	0	8
Multimodal Egocentric Activity [136]	Daily activities	✓	✓	RGB, sensor data	AR	2016	30,000	200	0.25 min	20	0	0	20
GUN-71 [137]	Daily activities	✓	✓	RGB, depth	AR	2015	12,000	N/A	N/A	71	28	0	8
Wearable Computer Vision System [135]	Daily activities	✓	✓	RGB, depth	AR	2014	N/A	5	N/A	12	0	0	4
NTU RGB+D 120 [132]	General	X	✓	RGB, depth	AR	2020	8 M	114,480	N/A	120	0	0	106
UTD-MHAD [129]	General	X	✓	RGB, depth, sensor data	AR	2017	N/A	861	N/A	27	0	0	8
SYSU 3D Human-Object Interaction [131]	General	X	✓	RGB, depth	AR	2017	N/A	480	N/A	12	0	0	40
UWA3D Multiview Activity [130]	General	X	✓	RGB, depth	AR	2016	N/A	1200	N/A	30	0	0	10
CAD-120 [128]	General	X	✓	RGB, depth	AR, AOR	2013	61585	120	0.28 min	10	12	N/A	4
MSRDailyActivity3D [126]	Daily activites	X	✓	RGB, depth	AR	2012	N/A	320	N/A	16	0	0	N/A
Human Activity Detection [127]	Daily activites	X	✓	RGB, depth	AR	2011	N/A	N/A	0.75 min	12	0	0	4
MSR-Action3D [125]	General	X	✓	depth	AR	2010	23797	402	0.07 min	20	0	0	7

Table 2.2: Comparative overview of relevant multimodal datasets. HOI: HOI Detection. EHOI: EHOI Detection. AR: Action Recognition. AOD: Active Object Detection. AOR: Active Object Recognition. OD: Object Detection, LOC: Localization, HPE: Hand Pose Estimation, NAO: Next-active objects detection.

which comprises both gaze and depth streams. The dataset has been acquired by 19 participants which performed 4 activities (indoor navigation, ball catching, object search and tea making). They used the Pupil Labs eye tracker to acquire the gaze signal, the MPU to obtain the IMU data and the ZED stereo camera to acquire the depth map.

Table 2.2 compares the aforementioned datasets with respect to the proposed MECCANO Multimodal dataset described in Section ??. The dataset is a conspicuous extension of the previous MECCANO dataset (see Section 4.3) and represents the first egocentric multimodal dataset comprising both gaze and depth signals, acquired in an industrial-like domain and annotated to tackle the next-active object detection task.

2.2.4 Next-Active Objects

The detection of next-active objects, i.e., the objects which will be involved in a EHOI, is a problem which has not been thoroughly studied due to the small number of public egocentric datasets suitable for the task. There are not egocentric datasets specifically annotated to perform this task. The authors of [139] have been the first to explore the next-active objects prediction problem. They performed experiments on the Activity of Daily Living (ADL) egocentric dataset, analyzing the trajectories of the next-active objects with a temporal sliding window. In [140], the task of anticipating egocentric actions has been addressed. The proposed architecture is composed by a motor attention module to predict the trajectory of the hands and by a module to detect the area of contact of the target object which will be active. These two outputs are fed into an anticipation module which predicts a spatio-temporal attention maps for the next interaction. The output is composed of the next action label, the hotspot which indicates the area of the object where there will be a contact and the hand trajectory. The two egocentric datasets ADL [123] and EPIC-Kitchens [119] have been re-annotaded by the authors of [141] to tackle the problem of short-term next-active object detection. They proposed a novel human-centerd approach composed by two pathways: 1) the first pathway generates a human visual attention probability map and 2) the other generates a human hand position probability map. These two maps are then fused by an interaction module which outputs the final map of the next-active object. In addition to the next-object location, the authors of [142] predicted the hands location in future frames. They tackled the problem designing a two-stream CNN architecture with an auto-encoder by extending a state-of-the-art convolutional object detection network (SSD) and using a regression network to infer future representations. The authors of [143] performed action anticipation and prediction through hand-objects contact representations. They presented a new architecture composed by an anticipation module and temporal relations represented using a Graph Convolutional Networks (GCN) and the LSTMs to predict the final next action label. They treated the next-active objects involved into the contact between hands and target objects through semantic segmentation masks.

Despite the problem of next-active object detection has been considered by these works, it has not yet been studied in depth also due to the absence of annotated egocentric datasets public available. In this thesis, we addressed the next-active object task on the MECCANO Multimodal datasets, which has been acquired in an industrial domain and annotated explicitly to tackle this challenging task.

Chapter 3

Visitors Behavioral Understanding

In this chapter we study the behavior of visitors in cultural sites which allows to build a wearable system able of assisting the visitor and collect useful information for the management of the site. We acquired two egocentric datasets due to the absence of public available first person vision data in the state-of-the-art. The collected datasets are described in Section 3.2 and Section 3.3. Together with the dataset, we proposed 5 fundamental tasks related to visitor behavior understanding, which can be addressed using the proposed dataset. In particular, we tackled the problem of *visitors localization*, treated in Section 3.4, the problem of the *detection and recognition of points of interest* in cultural sites which is discussed in Section 3.5 as well as the tasks of *object retrieval* and *survey prediction* which have been described in Section 3.6 and Section 3.7 respectively. We deeply explored the tasks of object detection and recognition considering the *semantic object segmentation* problem in cultural sites using synthetic and real data. We discuss this problem in Section 3.8. Finally, Section 3.9 presents the *VEDI System*, which is the final integrated wearable system developed to assist the visitors of cultural sites.

3.1 Assistance for the Human in Cultural Sites

Cultural sites receive lots of visitors every day. To improve the fruition of cultural objects, a site manager should be able to assist the users during their visit by providing additional information and suggesting what to see next, as well as to gather information to understand the behavior of the visitors (e.g., what has been liked most) in order to improve the suggested visit paths or the placement of artworks. Traditional ways to achieve such goals include the employment of professional guides, the installation of informative panels, the distribution of printed material to the users (e.g., maps and descriptions) and the collection of visitors' opinions through surveys. When the number of visitors grows large, the aforementioned traditional tools tend to become less effective, which motivates the employment of automated technologies. In order to assist visitors in a scalable and interactive way, site managers have employed technologies aimed to provide complementary information on the cultural objects on demand. An example of such technologies are audio guides, which allow to obtain spoken information about a point of interest by dialling the appropriate number on the device. Similarly, the use of tablets or smartphones allows to obtain audio-visual complementary information of an observed object of the cultural site by interacting with a touch interface (e.g., inserting the number of the cultural object of interest) or by taking a picture of a QR Code. Although effective in some cases, the aforementioned technologies are very limited by the following factors:

- they require the active intervention of the visitor, who needs to specify the correct number or to take a picture of the right QR Code;
- they require the site manager to install informative panels reporting the number or QR Code corresponding to a given cultural object (which is sometimes not possible due to the nature of the site).
- some technologies rely on the Internet, which is not always available in the cultural site (e.g., in an outdoor site or in caves).

Moreover, traditional systems are unable to acquire any information useful to understand the visitor's habits or interests (see Figure 3.1 - left). To gather information about the visitors (i.e., what they see and where they are) in an automated way, past works have employed fixed cameras and classic "third person vision" algorithms to detect, track, count people and estimate their gaze [27]. However, systems based on third person vision are capped by several limitations: 1) fixed cameras need to be installed in the cultural site and this is not always possible, 2) the fixed viewpoint of third person cameras makes it difficult to estimate what the visitors are looking at (e.g., ambiguity on estimation of what people see), 3) fixed cameras are easily affected by occlusion and people re-identification problems (e.g., difficulties to follow a person from a room to another), 3) the system has to work for several visitors at a time, making it difficult to profile them and to adapt its functioning to their specific needs (e.g., personal recommendation). Moreover, systems based on third person vision cannot easily communicate to the visitor in order to "augment his visit" providing information on the observed cultural object or by recommending what to see next.

Ideally, we would like to provide the user with an unobtrusive wearable device capable of addressing both tasks: augmenting the visit and inferring behavioral information about the visitors. We would like to note that wearable devices are particularly suited to solve this kind of tasks as they are naturally worn and carried by the visitor (see Figure 3.1 - right). Moreover, wearable systems do not require the explicit intervention of the visitor to deliver services such as localization, augmented reality and recommendation. The device should be aware of the current location of the visitor and capable to infer what he is looking at, and, ultimately, his behavior (e.g., what has already been seen? for how long?). Such a system would allow to provide automatic assistance to the visitor by showing him the current location, guiding him to a given area of the site, giving information about the currently observed cultural object, keeping track of what has been already seen and for how long, and suggesting what is yet to be seen by the visitor. Equipping multiple visitors with an egocentric vision device, it would be possible to track a profile of the different visitors in order to provide: 1) recommendations on what to see in the cultural site based on what has already been seen/liked (e.g, considering how much time has been spent at a given location or for how long the user has observed a cultural object), 2) statistics on the behaviors of the visitors within the site. Such statistics could be of great use by the site manager to improve the services offered by the cultural site and to facilitate the fruition of the cultural site.

As investigated by other authors [17, 18, 19, 22], wearable devices equipped with a camera such as smart glasses (e.g., Google Glass, Microsoft HoloLens and Magic Leap) offer interesting opportunities to develop the aforementioned technologies and services for visitors and site managers. Wearable glasses equipped with mixed reality visualization systems (i.e., capable of displaying virtual elements on images coming from the real world) such as Microsoft HoloLens and Magic Leap allow to provide information to the visitor in a natural way, for example by showing a 3D reconstruction of a cultural object or by showing virtual textual information next to a work of art. In particular, a wearable system should be able to carry out at

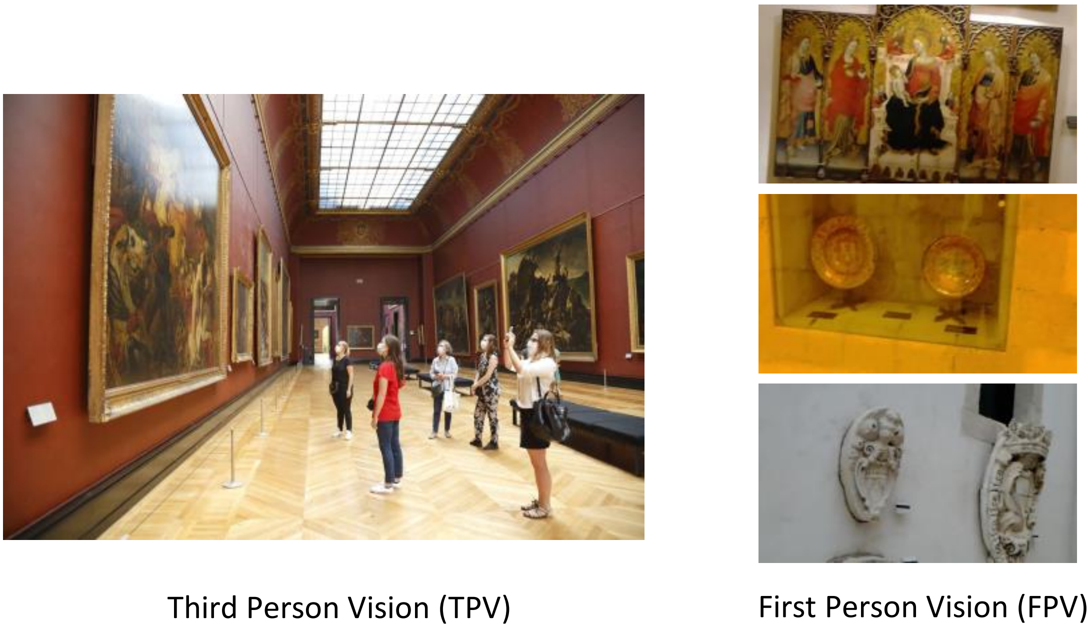


Figure 3.1: Comparison between the third person (left) and first person (right) vision related to the cultural heritage domain.

least the following tasks: 1) localizing the visitor at any moment of the visit, 2) recognizing the cultural objects observed by the visitor, 3) estimating the visitor's attention, 4) profiling the user, 5) recommending what to see next.

In this thesis, we describe VEDI (Vision Exploitation for Data Interpretation), an integrated system which includes a wearable device capable of supporting the visitors of cultural sites, as well as a back-end to analyze the visual information collected by the wearable system and infer behavioral information useful for the site manager (Section 3.9).

Points of Interest

In this section we describe the difference between a general object and a point of interest considering the domain of cultural heritage.

A point of interest can be defined by the site manager as an entity (e.g. object, architectural element, environment etc.) for which it is interesting to estimate the attention of visitors. Points of interest of a cultural site are those elements which are usually provided with information such that the visitors can understand what

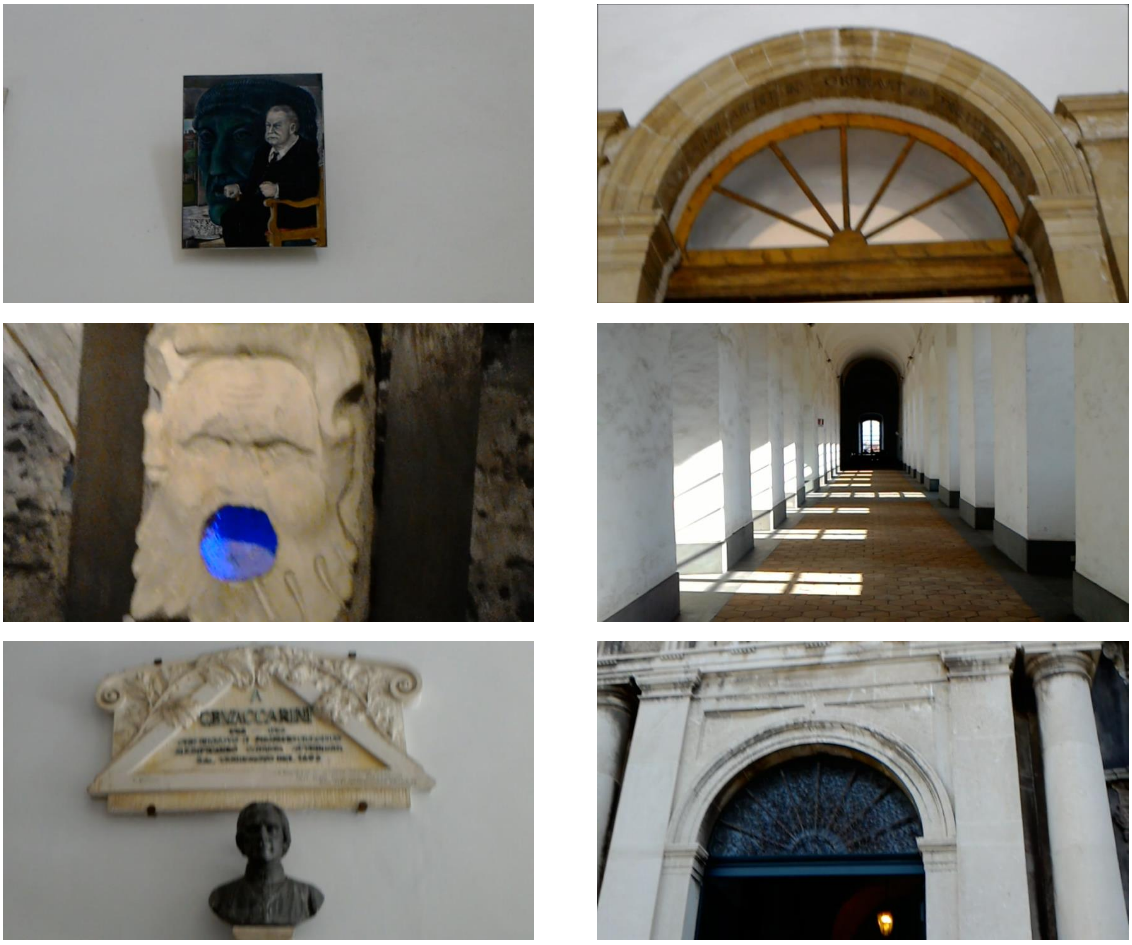


Figure 3.2: Some examples of points of interest: paintings, environments, statues and more. Note that the exhibited variability makes recognition hard.

they are observing. As such, it can be an object or an area of a environment, which increases variability in the recognition. Figure 3.2 shows some examples of points of interest such as paintings, environments or statues.

Past works have investigated the problem of estimating the attention of visitors from fixed cameras. However, this setup raises uncertainty about which object the user is looking at when there are more neighbouring objects. Figure 3.3 shows the constraints related to third person vision with respect to this task. As shown in the figure, there is ambiguity in understanding what the visitors are looking at (left image) and sometimes the point of interest observed by the user is out of the scene (right image), due to the unconvenient position of the fixed camera.

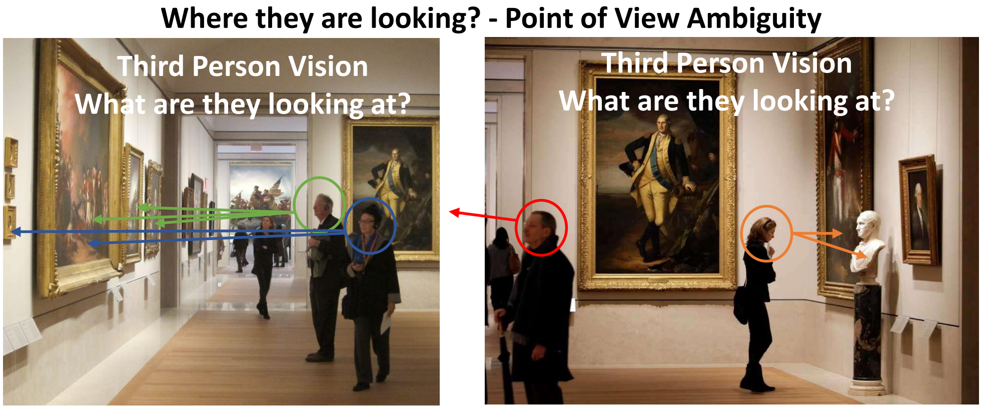

Where they are looking? - Point of View Ambiguity
Third Person Vision
What are they looking at?
Third Person Vision
What are they looking at?

Figure 3.3: The figure shows the constraints of using fixed cameras to infer the attention of the visitors, such as ambiguity on what the users see (on the left) and missing objects falling out of the scene (on the right).

In this thesis we address the problem of object detection in cultural sites considering the dual nature of the points of interest, which include objects and environments (Section 3.5).

3.2 UNICT-VEDI Dataset

We collected UNICT-VEDI [144], a large dataset of videos acquired in the *Monastero dei Benedettini*[1] cultural site, located in Catania, Italy. The dataset has been acquired using two wearable devices: a head-mounted Microsoft HoloLens[2] and a chest-mounted GoPro Hero4[3] (as shown in Figure 3.4) and it is publicly available at: `https://iplab.dmi.unict.it/VEDI/`. The considered devices represent two popular choices for the placement of wearable cameras.

Indeed, chest-mounted devices generally allow to produce better quality images due to the reduced egocentric motion with respect to head-mounted devices. On the other side, head-mounted cameras allow to capture what the user is looking at and hence they are better suited to model the attention of the user. Moreover, head-mounted devices such as HoloLens allow for the deployment of augmented reality,

[1] http://monasterodeibenedettini.it/

[2] https://www.microsoft.com/it-it/hololens

[3] https://gopro.com/it/it/update/hero4

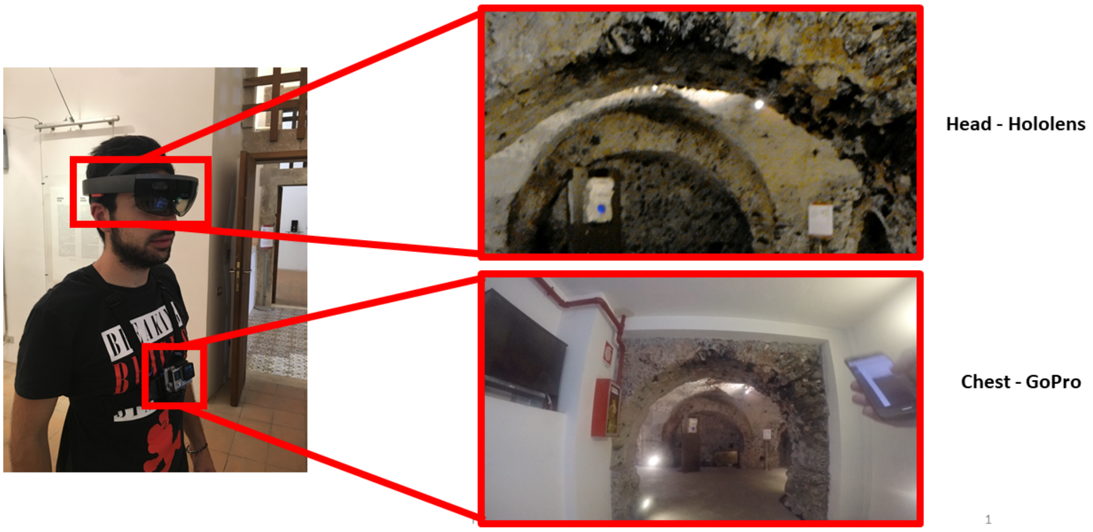


Figure 3.4: The figure shows the head-mounted Microsoft HoloLens and the chest-mounted GoPro Hero 4 set used to acquire the dataset.

which can be useful to assist the visitor of a cultural site. The two devices are also characterized by two different Fields Of View (FOV). In particular, the FOV of the HoloLens device is narrow-angle, while the FOV of the GoPro device is wide-angle. This is shown in Figure 3.5, which reports some frames acquired with HoloLens along with the corresponding images acquired with GoPro. In order to study which device is better suited to address the localization problem, we use the two devices simultaneously during the data acquisition procedure to collect two separate and compliant datasets: one containing only data acquired with HoloLens and the other one containing only data acquired using the GoPro device. The videos captured using the HoloLens device have a resolution of to 1216 × 684 pixels and a frame rate of 24 fps, whereas the videos recorded with GoPro Hero 4 have a resolution of 1280 × 720 pixels and a frame rate of 25 fps.

Each video frame has been labeled according to: 1) the location of the visitor and 2) the "point of interest" (i.e. the cultural object) observed by the visitor, if any. In both cases, a frame can be labeled as belonging to a "negative" class, which

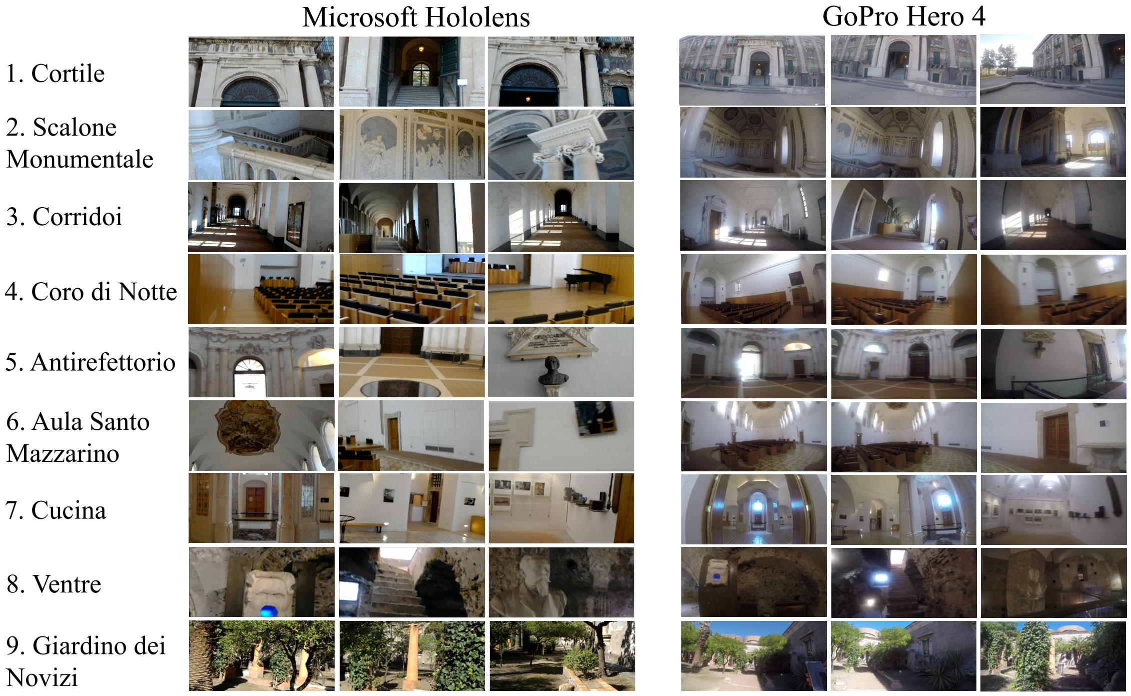


Figure 3.5: The figure shows some frames for each considered environment, acquired with Microsoft Hololens (left column) and GoPro Hero4 (right column) wearable devices.

denotes all visual information which is not of interest. For example, a frame is labeled as negative when the visitor is transiting through a corridor which has not been included in the training set because it is not a room (context) of interest for the visitors or when he is not looking at any of the considered points of interest. We considered a total of 9 environments and 57 points of interests. Each environment is identified by a number from 1 to 9, while points of interest (i.e., cultural objects) are denoted by a code in the form $X.Y$ (e.g., 2.3), where X denotes the environment in which the points of interest are located and Y identify the point of interest. Figure 3.5 shows some representative examples of each of the 9 considered environments. Table 3.1 shows the list of the considered environments (left column) and the related points of interest (right column). In the case of class *Cortile*, the same video is used to represent both the environment (1) and the related point of interest (Ingresso - 1.1). Figure 3.6 shows some representative examples of the 57 points of interest acquired with HoloLens and GoPro Hero 4, whereas Figure 3.7 reports some sample frames belonging to negative locations and points of interest. As can be noted from the reported samples, the GoPro device allow to acquire a larger amount of visual information, due to its wide-angle field of view. On the contrary, data acquired using the HoloLens device tends to exhibit more visual variability, due to the head-mounted point of view, which suggests its better suitability for the recognition of objects of interest and behavioral understanding.

The dataset is composed of separate training and test videos, which have been collected following two different protocols. To collect the training set, we acquired a set of videos (at least one) for each of the considered environments and a set of videos for each of the considered points of interest. Environment-related training videos have been acquired by an operator who had been instructed to walk into the environment and look around to capture images from different points of view. A similar procedure has been employed to acquire training videos for the different points of interest. In this case, the operator has been instructed to walk around the object and look at it from different points of view. For each camera, we collected a total of 12 training videos for the 9 environments and 68 videos for the 57 points of interest. This accounts to a total of 80 training videos for each camera. Table 3.2 summarizes the number of training videos acquired for each environment.

The test videos have been acquired by operators who have been asked to simulate

Table 3.1: The table reports the list of all environments and the related points of interest contained. In parenthesis, we report the unique numerical code of the environment/point of interest.

Environments	**Points of Interest**
Cortile (1)	Ingresso (1.1)
Scal. Monumentale (2)	RampaS.Nicola (2.1)
	RampaS.Benedetto (2.2)
Corridoi (3)	SimboloTreBiglie (3.1)
	ChiostroLevante (3.2)
	Plastico (3.3)
	Affresco (3.4)
	Finestra_ChiostroLev. (3.5)
	PortaCorodiNotte (3.6)
	TracciaPortone (3.7)
	StanzaAbate (3.8)
	CorridoioDiLevante (3.9)
	CorridoioCorodiNotte (3.10)
	CorridoioOrologio (3.11)
Coro di Notte (4)	Quadro (4.1)
	PavimentoOrig.Altare (4.2)
	BalconeChiesa (4.3)
Antirefettorio (5)	PortaAulaS.Mazzarino (5.1)
	PortaIng.MuseoFabb. (5.2)
	PortaAntirefettorio (5.3)
	PortaIngressoRef.Piccolo (5.4)
	Cupola (5.5)
	AperturaPavimento (5.6)
	S.Agata (5.7)
	S.Scolastica (5.8)
	ArcoconFirma (5.9)
	BustoVaccarini (5.10)
Aula S.Mazzarino (6)	Quadro S.Mazzarino (6.1)
	Affresco (6.2)
Aula S.Mazzarino (6)	PavimentoOriginale (6.3)
	PavimentoRestaurato (6.4)
	BassorilieviMancanti (6.5)
	LavamaniSx (6.6)
	LavamaniDx (6.7)
	TavoloRelatori (6.8)
	Poltrone (6.9)
Cucina (7)	Edicola (7.1)
	PavimentoA (7.2)
	PavimentoB (7.3)
	PassavivandePav.Orig. (7.4)
	AperturaPavimento (7.5)
	Scala (7.6)
	SalaMetereologica (7.7)
Ventre (8)	Doccione (8.1)
	VanoRaccoltaCenere (8.2)
	SalaRossa (8.3)
	ScalaCucina (8.4)
	CucinaProvv. (8.5)
	Ghiacciaia (8.6)
	Latrina (8.7)
	OssaeScarti (8.8)
	Pozzo (8.9)
	Cisterna (8.10)
	BustoPietroTacchini (8.11)
Giardino Novizi (9)	NicchiaePavimento (9.1)
	TraccePalestra (9.2)
	PergolatoNovizi (9.3)

Table 3.2: Training videos and total number of frames for each environment.

Environment	**#Videos**	**#Frames**
1 Cortile	1	1171
2 Scalone Monumentale	6	13464
3 Corridoi	14	31037
4 Coro di Notte	7	12687
5 Antirefettorio	14	29918
6 Aula Santo Mazzarino	10	31635
7 Cucina	10	31112
8 Ventre	14	68198
9 Giardino dei Novizi	4	10852
Total	80	230074

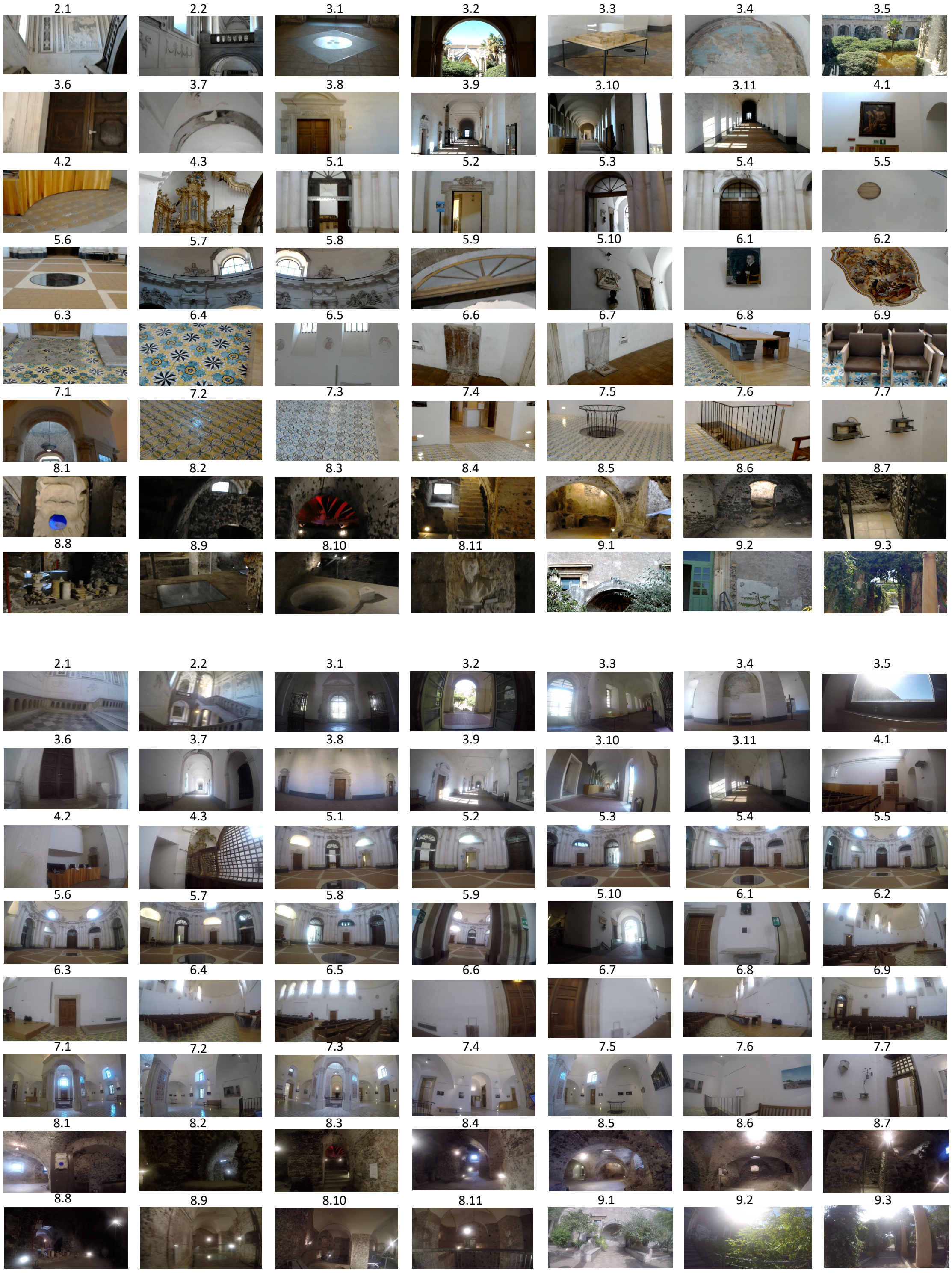


Figure 3.6: The figure shows a sample frame for each of the 57 points of interest acquired with both Microsoft Hololens (top) and GoPro (bottom).

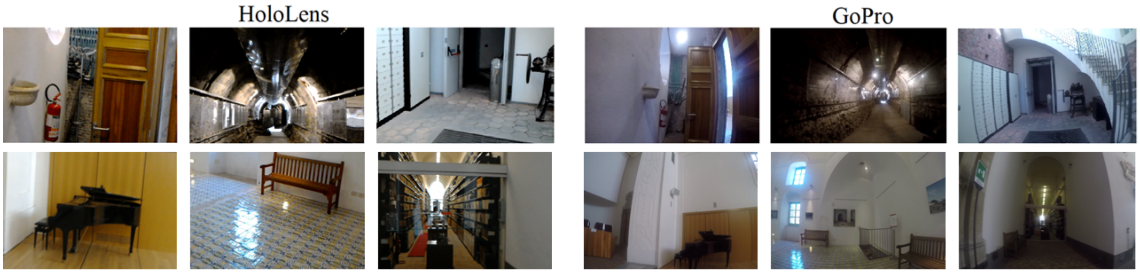


Figure 3.7: Example of frames belonging to the negative class, acquired with Microsoft Hololens (left) and GoPro (right).

a visit to the cultural site. No specific information on where to move, what to look at, and for how long have been given to the operators. Test videos have been acquired by three different subjects. Specifically, we acquired 7 test video per wearable camera, totaling 14 test videos. Each HoloLens video is composed by one or more video fragments due to the video recording time limits imposed by the default Hololens video capture application. Table 3.3 reports the list of test videos acquired using HoloLens and GoPro. For each test video, we report the number of frames and the list of environments the user visited during the acquisition of the video.

3.2.1 Microsoft Hololens Data

Table 3.4 details the training videos acquired with the Microsoft Hololens device to represent each of the considered environments. For each class, we report the total duration of the videos, the required storage, the number of frames and the percentage of frames, with respect to the total number of frames in the training set. Table 3.5 details the training videos acquired to represent each of the considered points of interest. For each class, we report the total duration of the videos, the required storage, the number of frames and the percentage of frames, with respect to the total number of frames in the training set. Table 3.6 reports a list of all test videos acquired using HoloLens. For each video, we report: Time, Storage, number of frames and percentage of frames with respect to the total number of frames of the training dataset.

Table 3.3: The list of test videos acquired using HoloLens (left) and GoPro (right). For each video, we report the number of frames and the list of environments visited by the user during the acquisition. The last rows report the total number of frames.

HoloLens		
Name	**#Frames**	**Environments**
Test1.0	7202	1 - 2 - 3 - 4 - 5 - 6
Test1.1	7202	
Test2.0	7203	1 - 2 - 3 - 4
Test3.0	7202	1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9
Test3.1	7203	
Test3.2	7201	
Test3.3	7202	
Test3.4	5694	
Test4.0	7204	
Test4.1	7202	1 - 2 - 3 - 4 - 5 - 7 - 8 - 9
Test4.2	3281	
Test4.3	7202	
Test4.4	4845	
Test5.0	6590	1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9
Test5.1	7202	
Test5.2	7202	
Test5.3	7201	
Test6.0	7202	1 - 2 - 3
Test7.0	7202	1 - 2 - 3 - 5
Test7.1	2721	
Total	131163	

GoPro		
Name	**#Frames**	**Environments**
Test1	14788	1 - 2 - 3 - 4
Test2	10503	1 - 2 - 3 - 5
Test3	14491	1 - 2 - 3 - 5 - 9
Test4	36808	1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9
Test5	18788	1 - 2 - 3 - 5 - 7 - 8
Test6	12661	2 - 3 - 4 - 8 - 9
Test7	38725	1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9
Total	146764	

Table 3.4: List of training videos of environments acquired using Hololens. For each video we show: time, amount of occupied memory, number of frames, percentage of frames with respect to the total number of frames of the training set.

Name	Time (s)	Storage (MB)	#frames	%frames
1.0_Cortile	48	48.145	1171	0,24%
2.0_Scalone	81	80.085	1947	0,40%
2.0_Scalone1	116	113.173	2747	0,56%
2.0_Scalone2	38	37.758	917	0,19%
4.0_CoroDiNotte	169	167.614	4068	0,83%
4.0_CoroDiNotte1	44	44.068	1067	0,22%
5.0_Antirefettorio	247	244.500	5933	1,21%
5.0_Antirefettorio1	263	260.395	6315	1,29%
6.0_SantoMazzarino	241	239.007	5800	1,18%
7.0_Cucina	239	237.268	5753	1,17%
8.0_Ventre	679	832.844	20385	4,15%
9.0_GiardinoNovizi	116	113.949	2788	0,57%
AVG	175.46	201.57	4908	1,00%

Table 3.5: List of training videos of points of interest acquired with Hololens. For each video we show: Time, Storage, number of frames and percentage of frames with respect to the total number of frames of the training dataset. The table continues in the next page.

Name	Time (s)	Storage (MB)	#frame	%frame
2.1_Scalone_RampaS.Nicola	163	161.776	3926	0,80%
2.2_Scalone_RampaS.Benedetto	14	14.471	354	0,07%
2.2_Scalone_RampaS.Benedetto1	148	147.226	3573	0,73%
3.1_Corridoi_TreBiglie	75	74.261	1804	0,37%
3.2_Corridoi_ChiostroLevante	55	54.500	1328	0,27%
3.3_Corridoi_Plastico	80	79.179	1927	0,39%
3.4_Corridoi_Affresco	74	74.023	1800	0,37%
3.5_Corridoi_Finestra_ChiostroLev.	84	83.102	2035	0,41%
3.6_Corridoi_PortaCoroDiNotte	53	53.255	1291	0,26%
3.6_Corridoi_PortaCoroDiNotte1	60	59.411	1446	0,29%
3.6_Corridoi_PortaCoroDiNotte2	58	57.901	1406	0,29%
3.7_Corridoi_TracciaPortone	53	53.118	1291	0,26%
3.8_Corridoi_StanzaAbate	81	79.773	1948	0,40%

3.9_Corridoi_CorridoioDiLevante	89	88.173	2142	0,44%
3.10_Corridoi_CorridoioCorodiNotte	122	121.321	2945	0,60%
3.10_Corridoi_CorridoioCorodiNotte1	103	101.796	2472	0,50%
3.11_Corridoi_CorridoioOrologio	300	296.488	7202	1,47%
4.1_CoroDiNotte_Quadro	71	70.245	1716	0,35%
4.2_CoroDiNotte_Pav.Orig.Altare	73	72.358	1759	0,36%
4.3_CoroDiNotte_BalconeChiesa	39	39.331	957	0,19%
4.3_CoroDiNotte_BalconeChiesa1	49	48.550	1185	0,24%
4.3_CoroDiNotte_BalconeChiesa2	80	79.861	1935	0,39%
5.1_Antirefettorio_PortaA.S.Mazz.Ap.	67	67.001	1630	0,33%
5.1_Antirefettorio_PortaA.S.Mazz.Ch.	74	73.589	1785	0,36%
5.2_Antirefettorio_PortaMuseoFab.Ap.	50	49.840	1211	0,25%
5.2_Antirefettorio_PortaMuseoFab.Ch.	62	61.846	1503	0,31%
5.3_Antirefettorio_PortaAntiref.	51	58.557	1537	0,31%
5.4_Antirefettorio_PortaRef.Piccolo	54	53.767	1306	0,27%
5.5_Antirefettorio_Cupola	57	56.089	1377	0,28%
5.6_Antirefettorio_AperturaPavimento	55	54.586	1322	0,27%
5.7_Antirefettorio_S.Agata	48	47.820	1165	0,24%
5.8_Antirefettorio_S.Scolastica	58	57.919	1407	0,29%
5.9_Antirefettorio_ArcoconFirma	62	76.700	1864	0,38%
5.10_Antirefettorio_BustoVaccarini	65	64.298	1563	0,32%
6.1_SantoMazzarino_QuadroS.Mazz.	71	70.401	1716	0,35%
6.2_SantoMazzarino_Affresco	213	211.424	5124	1,04%
6.3_SantoMazzarino_PavimentoOr.	99	98.691	2397	0,49%
6.4_SantoMazzarino_PavimentoRes.	69	69.148	1675	0,34%
6.5_SantoMazzarino_BassorilieviManc.	117	115.348	2823	0,57%
6.6_SantoMazzarino_LavamaniSx	151	149.882	3637	0,74%
6.7_SantoMazzarino_LavamaniDx	93	92.928	2256	0,46%
6.8_SantoMazzarino_TavoloRelatori	150	148.661	3603	0,73%
6.9_SantoMazzarino_Poltrone	108	107.374	2604	0,53%
7.1_Cucina_Edicola	369	437.219	11086	2,26%
7.2_Cucina_PavimentoA	52	52.163	1268	0,26%
7.3_Cucina_PavimentoB	52	52.244	1266	0,26%

7.4_Cucina_PassavivandePavim.Orig.	81	80.733	1961	0,40%
7.5_Cucina_AperturaPav.1	57	57.170	1385	0,28%
7.5_Cucina_AperturaPav.2	53	52.875	1280	0,26%
7.5_Cucina_AperturaPav.3	62	62.320	1509	0,31%
7.6_Cucina_Scala	77	76.587	1856	0,38%
7.7_Cucina_SalaMetereologica	156	154.394	3748	0,76%
8.1_Ventre_Doccione	103	102.683	2492	0,51%
8.2_Ventre_VanoRacc.Cenere	126	124.837	3026	0,62%
8.3_Ventre_SalaRossa	300	296.792	7202	1,47%
8.4_Ventre_ScalaCucina	214	212.372	5152	1,05%
8.5_Ventre_CucinaProvv.	148	146.379	3553	0,72%
8.6_Ventre_Ghiacciaia	69	68.562	1668	0,34%
8.6_Ventre_Ghiacciaia1	266	263.817	6398	1,30%
8.7_Ventre_Latrina	102	100.542	2468	0,50%
8.8_Ventre_OssaScarti	154	152.861	3713	0,76%
8.8_Ventre_OssaScarti1	36	36.345	886	0,18%
8.9_Ventre_Pozzo	300	297.167	7209	1,47%
8.10_Ventre_Cisterna	57	56.572	1384	0,28%
8.11_Ventre_BustoP.Tacchini	110	109.520	2662	0,54%
9.1_GiardinoNovizi_NicchiaePav.	106	105.004	2555	0,52%
9.2_GiardinoNovizi_TraccePalestra	63	62.464	1525	0,31%
9.3_GiardinoNovizi_Pergolato	166	164.150	3984	0,81%
AVG	102.6	103.26	2517	0,51%

3.2.2 GoPro Data

Table 3.7 details the training videos acquired using the GoPro device to represent each of the considered environments. For each class, we report the total duration of the videos, the required storage, the number of frames and the percentage of frames, with respect to the total number of frames in the training set. Table 3.8 details the training videos acquired to represent each of the considered points of interest. For each class, we report the total duration of the videos, the required storage, the

Table 3.6: List of test videos acquired using HoloLens. For each video we report the length in seconds, the amount of occupied memory, the number of frames, the number of environments included in the video, the number of points of interest included in the video, the sequence of environments as navigated by the visitor, and the sequence of points of interest, as navigated by the visitor.

Video	Time	MB	#frames	%frames	#env.	#p.int.	seq.environments	seq. p.interest
Test1	300	296.896	7202	5,49%	4	10	1->2->3->4	1.1->2.2->3.1->3.9->3.10->3.4->3.6->4.1->4.2->4.3
Test1.1	300	297.031	7202	5,49%	4	11	4->3->5->6	4.3->3.11->5.6->5.7->5.1->6.1->6.4->6.9->6.3->6.6->6.2
Test2.0	300	296.993	7203	5,49%	4	9	1_>2->3->4	1.1->2.1->3.9->3.10->3.5->3.10->3.5->3.10->3.6->4.1->4.2->4.3
Test3.0	300	296.959	7202	5,49%	4	8	1->2->3->4	1.1->2.1->3.9->3.10->3.4->3.11->3.6->4.1
Test3.1	300	296.913	7203	5,49%	3	5	4->3->5	4.2->4.3->4.2->3.11->5.5->5.6
Test3.2	300	297.083	7201	5,49%	4	17	5->6->5->7	5.6->5.1->5.3->5.10->5.9->5.1->6.2->6.1->6.4->6.9->6.5->6.4->6.3->6.8->6.5->6.7->6.6->6.5->5.2->7.1
Test3.3	300	297.160	7202	5,49%	2	12	7->8	7.2.>7.6->7.5->7.4->7.7->7.1->8.1->8.6->8.2->8.8->8.4->8.10
Test3.4	237	234.744	5694	4,34%	3	5	8->9->3	8.9->8.11->8.3->9.1->9.2
Test4.0	300	296.979	7204	5,49%	4	10	1->2->3->5	1.1->2.2->3.9->3.10->3.4->3.11->->3.7->3.11->5.10->5.9->5.1
Test4.1	300	296.654	7202	5,49%	3	16	5->7->8	5.6->5.5->5.8->5.4->5.2->7.1->7.3->7.2->7.1->7.4->7.1->7.5->7.7->7.1->8.1->8.5->8.2->8.4->8.7
Test4.2	136	135.143	3281	2,50%	1	3	8	8.10->8.9->8.11
Test4.3	300	296.853	7202	5,49%	4	7	8->9->3->4	8.3->9.1->9.3->9.2->3.11->3.6->4.2
Test4.4	201	199.585	4845	3,69%	3	4	4->3->2	4.2->3.10->3.9->2.1
Test5.0	274	271.524	6590	5,02%	4	11	1->2->3->2->3->4	1.1->3.1->3.3->3.2->2.1->3.9->3.10->3.5->3.10->3.5->3.10->3.4->3.7->3.4->3.6
Test5.1	300	296.801	7202	5,49%	4	11	4->3->9->3->5	4.1->4.2->3.6->3.11->9.2->3.11->->5.5->5.3->5.9->5.7->5.8->5.2
Test5.2	300	296.921	7202	5,49%	2	13	7->8	7.1->7.2->7.3->7.1->7.5->7.1->7.1->7.4->7.6->7.7->7.4->7.7->8.1->8.5->8.4->8.5->8.6->8.2->8.8->8.4
Test5.3	300	297.063	7201	5,49%	4	17	8->7->5->6	8.11->8.9->8.3->8.4->8.5->8.6->7.1->7.5->7.2->5.1->6.2->6.4->6.1->6.5->6.3->6.5->6.2->6.7->6.9
Test6.0	300	296.880	7202	5,49%	3	9	1->2->3	1.1->2.2->2.1->3.9->3.10->3.4->3.11->3.6->3.11->3.7->3.11
Test7.0	300	296.945	7202	5,49%	3	7	1->2->3	1.1->2.2->3.9->3.10->3.4->3.6->3.11
Test7.1	113	112.030	2721	2,07%	3	3	3->5->3	3.11->5.9->5.10->3.11

Table 3.7: List of training videos of environments acquired using GoPro. For each video we show: time, amount of occupied memory, number of frames, percentage of frames with respect to the total number of frames of the training set.

Name	Time (s)	Storage (MB)	#frames	%frames
1.0_Cortile	47	116.798	1187	0,24%
2.0_Scalone	81	201.194	2044	0,42%
2.0_Scalone1	111	274.094	2785	0,57%
2.0_Scalone2	36	90.250	918	0,19%
4.0_CoroDiNotte	168	413.702	4205	0,86%
4.0_CoroDiNotte1	43	107.968	1097	0,22%
5.0_Antirefettorio	191	471.275	4790	0,97%
5.0_Antirefettorio1	262	644.852	6554	1,33%
6.0_SantoMazzarino	240	591.177	6009	1,22%
7.0_Cucina	241	592.907	6027	1,23%
8.0_Ventre	699	1719.352	17478	3,56%
9.0_GiardinoNovizi	116	427.472	6965	1,42%
AVG	186.25	242.62	5005	1,02%

number of frames and the percentage of frames, with respect to the total number of frames in the training set. Table 3.9 reports a list of all test videos acquired using GoPro. For each video, we report: Time, Storage, number of frames and percentage of frames with respect to the total number of frames of the training dataset.

Table 3.8: List of training videos of environments acquired using GoPro. For each video we show: time, amount of occupied memory, number of frames, percentage of frames with respect to the total number of frames of the training set.

Name	Time (s)	Storage (MB)	#frame	%frame
2.1_Scalone_RampaS.Nicola	160	394,844	4013	0,82%
2.2_Scalone_RampaS.Benedetto	14	35,758	363	0,07%
2.2_Scalone_RampaS.Benedetto1	147	361,856	3678	0,75%
3.1_Corridoi_TreBiglie	72	178,313	1812	0,37%
3.2_Corridoi_ChiostroLevante	54	202,165	3295	0,67%
3.3_Corridoi_Plastico	79	195,673	1989	0,40%
3.4_Corridoi_Affresco	74	182,106	1850	0,38%
3.5_Corridoi_Finestra_ChiostroLev.	81	300,474	4896	1,00%
3.6_Corridoi_PortaCoroDiNotte	50	123,34	1253	0,26%
3.6_Corridoi_PortaCoroDiNotte1	60	148,97	1514	0,31%

3.6_Corridoi_PortaCoroDiNotte2	58	142,747	1451	0,30%
3.7_Corridoi_TracciaPortone	55	136,484	1387	0,28%
3.8_Corridoi_StanzaAbate	58	122,859	1755	0,36%
3.9_Corridoi_CorridoioDiLevante	85	211,539	2148	0,44%
3.10_Corridoi_CorridoioCorodiNotte	120	297,325	3022	0,62%
3.10_Corridoi_CorridoioCorodiNotte1	100	246,769	2509	0,51%
3.11_Corridoi_CorridoioOrologio	176	434,081	4412	0,90%
4.1_CoroDiNotte_Quadro	69	171,551	1743	0,35%
4.2_CoroDiNotte_Pav.Orig.Altare	72	178,448	1813	0,37%
4.3_CoroDiNotte_BalconeChiesa	39	96,281	978	0,20%
4.3_CoroDiNotte_BalconeChiesa1	48	119,512	1214	0,25%
4.3_CoroDiNotte_BalconeChiesa2	80	197,29	2004	0,41%
5.1_Antirefettorio_PortaA.S.Mazz.Ap.	66	164,644	1673	0,34%
5.1_Antirefettorio_PortaA.S.Mazz.Ch	74	183,481	1864	0,38%
5.2_Antirefettorio_PortaMuseoFab.Ap	50	124,991	1270	0,26%
5.2_Antirefettorio_PortaMuseoFab.Ch	62	154,254	1567	0,32%
5.3_Antirefettorio_PortaAntiref.	52	128,695	1306	0,27%
5.4_Antirefettorio_PortaRef.Piccolo	54	133,494	1356	0,28%
5.5_Antirefettorio_Cupola	55	135,594	1378	0,28%
5.6_Antirefettorio_AperturaPavimento	57	141,647	1439	0,29%
5.7_Antirefettorio_S.Agata	49	120,611	1225	0,25%
5.8_Antirefettorio_S.Scolastica	56	137,79	1400	0,28%
5.9_Antirefettorio_ArcoconFirma	60	149,889	1522	0,31%
5.10_Antirefettorio_BustoVaccarini	66	162,799	1654	0,34%
6.1_SantoMazzarino_QuadroS.Mazz.	72	178,16	1810	0,37%
6.2_SantoMazzarino_Affresco	214	526,612	5353	1,09%
6.3_SantoMazzarino_PavimentoOr.	98	242,332	2462	0,50%
6.4_SantoMazzarino_PavimentoRes.	68	169,524	1722	0,35%
6.5_SantoMazzarino_BassorilieviMan.	116	25,942	2906	0,59%
6.6_SantoMazzarino_LavamaniSx	134	331,517	3369	0,69%
6.7_SantoMazzarino_LavamaniDx	91	225,892	2296	0,47%
6.8_SantoMazzarino_TavoloRelatori	149	366,575	3726	0,76%
6.9_SantoMazzarino_Poltrone	108	266,611	2709	0,55%

7.1_Cucina_Edicola	368	931,368	9467	1,93%
7.2_Cucina_PavimentoA	55	137,197	1394	0,28%
7.3_Cucina_PavimentoB	52	129,601	1316	0,27%
7.4_Cucina_PassavivandePavim.Orig.	84	206,667	2100	0,43%
7.5_Cucina_AperturaPav.1	58	142,7	1450	0,30%
7.5_Cucina_AperturaPav.2	57	142,145	1444	0,29%
7.5_Cucina_AperturaPav.3	65	161,61	1641	0,33%
7.6_Cucina_Scala	79	195,547	1988	0,40%
7.7_Cucina_SalaMetereologica	158	390,583	3970	0,81%
8.1_Ventre_Doccione	100	246,353	2503	0,51%
8.2_Ventre_VanoRacc.Cenere	128	315,824	3210	0,65%
8.3_Ventre_SalaRossa	158	388,603	3951	0,80%
8.4_Ventre_ScalaCucina	213	524,06	5327	1,08%
8.5_Ventre_CucinaProvv.	151	371,84	3779	0,77%
8.6_Ventre_Ghiacciaia	72	178,295	1811	0,37%
8.6_Ventre_Ghiacciaia1	185	457,1	4646	0,95%
8.7_Ventre_Latrina	80	295,294	4804	0,98%
8.8_Ventre_OssaScarti	150	369,523	3756	0,76%
8.8_Ventre_OssaScarti1	34	86,105	873	0,18%
8.9_Ventre_Pozzo	149	367,713	3737	0,76%
8.10_Ventre_Cisterna	30	114,114	1852	0,38%
8.11_Ventre_BustoP.Tacchini	106	261,822	2661	0,54%
9.1_GiardinoNovizi_NicchiaePav.	104	256,731	2609	0,53%
9.2_GiardinoNovizi_TraccePalestra	58	216,801	3533	0,72%
9.3_GiardinoNovizi_Pergolato	164	404,893	4115	0,84%
AVG	93.53	232.97	2515	0,51%

Table 3.9: List of test videos acquired using GoPro. For each video we report the length in seconds, the amount of occupied memory, the number of frames, the number of environments included in the video, the number of points of interest included in the video, the sequence of environments as navigated by the visitor, and the sequence of points of interest, as navigated by the visitor.

Video	Time(s)	MB	#frames	%frames	#env.	#p.int.	sequence environements	sequence points of interest
Test1	397	390.558	14788	5.32%	4	9	1.0->2.0->3.0->4.0->3.0	1.1->2.2->3.9->3.10->3.6->4.1->4.2->4.3->3.11
Test2	420	1033.149	10503	3.78%	4	8	1.0->2.0->3.0->5.0->3.0	1.1->2.1->3.9->3.10->3.4->3.11->3.6->3.11->5.10->3.10
Test3	579	1425.480	14491	5.21%	5	11	1.0->2.0->3.0->9.0->3.0->5.0->3.0	1.1->2.2->2.1->2.2->3.9->3.10->3.6->3.11->9.2->9.1->9.3->9.2->3.11->5.10->3.11
Test4	1472	3620.627	36808	13.24%	9	35	1.0->2.0->3.0->4.0->3.0->5.0->6.0->5.0->7.0->8.0->9.0->3.0	1.1->2.1->2.2->3.9->3.10->3.4->3.11->3.6->4.1->4.3->3.11->5.4->5.2->5.1->5.3->5.10->5.9->5.6->5.1->6.9->6.4->6.9->6.8->5.3->5.6->5.4->5.2->7.1->7.6->7.1->7.5->7.4->7.7->8.1->8.6->8.2->8.8->8.1->8.10->8.9->8.11->8.3->9.1->9.2
Test5	751	1848.142	18788	6.76%	6	26	1.0->2.0->3.0->5.0->7.0->8.0	1.1->2.2->2.1->3.1->3.9->3.10->3.4->3.11->5.10->5.6->5.1->5.3->5.4->5.2->7.1->7.4->7.1->7.5->7.7->8.1->8.5->8.4->8.7->8.10->8.8->8.10->8.11
Test6	506	1245.254	12661	4.56%	5	11	8.0->9.0->3.0->4.0->3.0->2.0	8.3->9.1->9.3->9.2->3.11->3.6->4.2->3.10->3.9->3.1->2.1
Test7	1549	3809.218	38725	13.93%	9	44	1.0->2.0->3.0->2.0->3.0->4.0->3.0->9.0->3.0->5.0->7.0->8.0->7.0->5.0->6.0->5.0->3.0->9.0->3.0->2.0->1.0	1.1->2.1->3.3->3.2->2.1->3.9->3.10->3.5->3.10->3.5->3.10->3.4->3.11->3.7->3.11->3.10->3.4->3.6->4.1->4.3->3.6->3.11->9.2->9.1->9.2->3.11->5.7->5.4->5.3->5.2->7.1->7.5->7.4->7.6->7.1->7.7->7.4->7.7->8.1->8.5->8.4->8.6->8.2->8.8->8.4->8.10->8.11->8.3->8.5->7.1->6.2->6.1->6.5->6.9->6.6->6.5->6.8->6.4->5.3->5.10->3.11->9.1->9.3->9.2->3.11->3.10->3.5->3.10->3.9->2.1->2.2->1.1

3.2.3 UNICT-VEDI Dataset for Points of Interest Recognition

To study the problem of Points of Interest Detection and Recognition in cultural sites, we extended the UNICT-VEDI dataset [145] annotating with bounding boxes the presence of 57 different points of interest in a subset of the frames of the dataset. The UNICT-VEDI-POIs dataset is publicly available at: `https://iplab.dmi.unict.it/VEDI_POIs/`. We only considered data acquired using the head-mounted Microsoft HoloLens device. For each of the 57 points of interest included in the UNICT-VEDI dataset, we annotated approximately 1,000 frames from the provided training videos, for a total of $54,248$ frames. Figure 3.8 shows some examples of the 57 points of interest annotated with bounding boxes.

The test videos have been sub-sampled at 1 frame per second and annotated

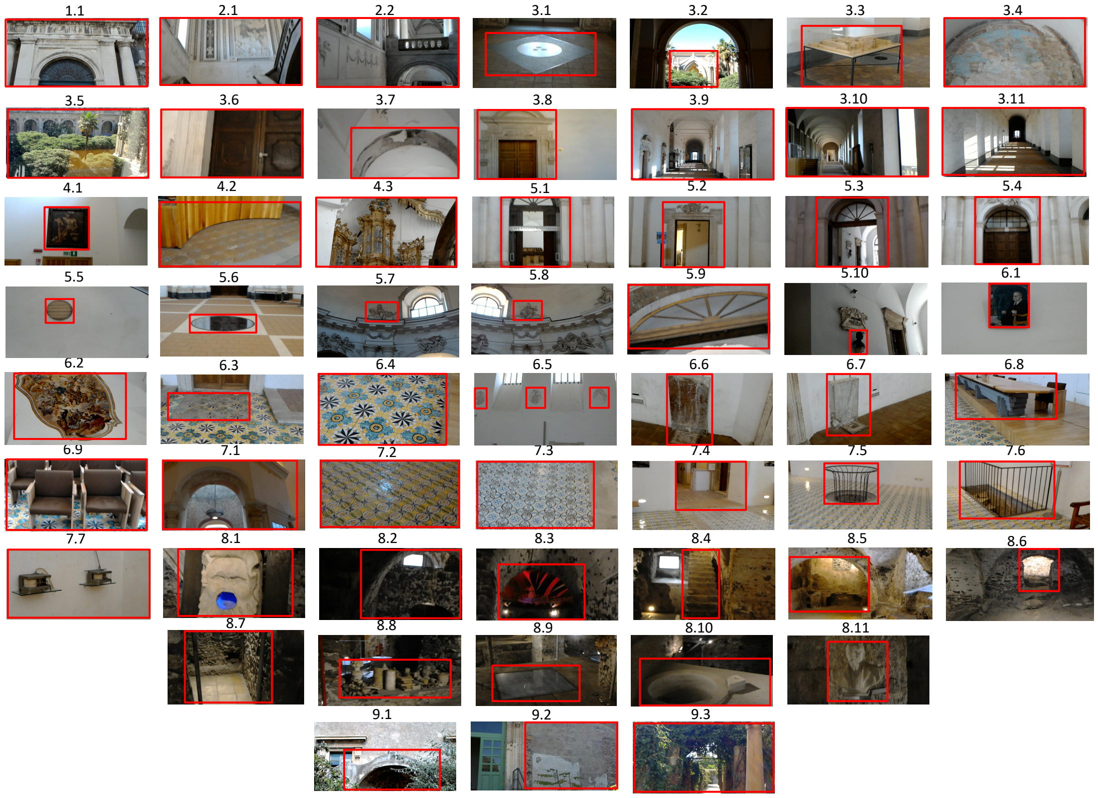


Figure 3.8: Sample frames with bounding box annotations related to the the 57 points of interest of the UNICT-VEDI dataset. Note that the annotations of some points of interest occupy the whole frame.

Table 3.10: Total number of frames (second column) and number of frames annotated with bounding boxes for each test video (third column) of the UNICT-VEDI dataset.

Name	**#frames**	**# frames with b_box**
Test1	14404	444
Test2	7203	220
Test3	41706	929
Test4	22530	767
Test5	28195	786
Test6	7202	231
Test7	9923	296
Total	**131163**	**3673**

with bounding boxes. Table 3.10 (third column) compares the number of frames annotated with bounding boxes for each test video with respect to the total numbers of frames (second column). A frame is labeled as "negative" if it does not contain any of the points of interest. Figure 3.9 shows the number of "negative" and "positive" frames belonging to the 57 points of interest for each test video. The number of "negative" frames demonstrates that the user often looks at something that is not a point of interest and therefore it is important to correctly reject these frames during the recognition procedure.

The distribution of labels (57 points of interest) in the 7 test videos is reported in Figure 3.10.

3.3 Egocentric Cultural Heritage (EGO-CH) Dataset

Due to the limited number of public egocentric datasets suitable to study the visitor's behavior, we acquired EGOcentric-Cultural Heritage (EGO-CH)[146], the first large dataset of egocentric videos for visitors behavioral understanding in cultural sites. It is publicly available at: `https://iplab.dmi.unict.it/EGO-CH/`. The dataset has been collected in two cultural sites located in Sicily, Italy: Galleria Regionale di Palazzo Bellomo[4] and Monastero dei Benedettini[5]. The overall dataset contains more than 27 hours of video, including 26 environments, over 200 Points of Interest and 70 visits. Please note that this set of data is adapted from and extends

[4] `http://www.regione.sicilia.it/beniculturali/palazzobellomo/`.

[5] `http://www.monasterodeibenedettini.it/`

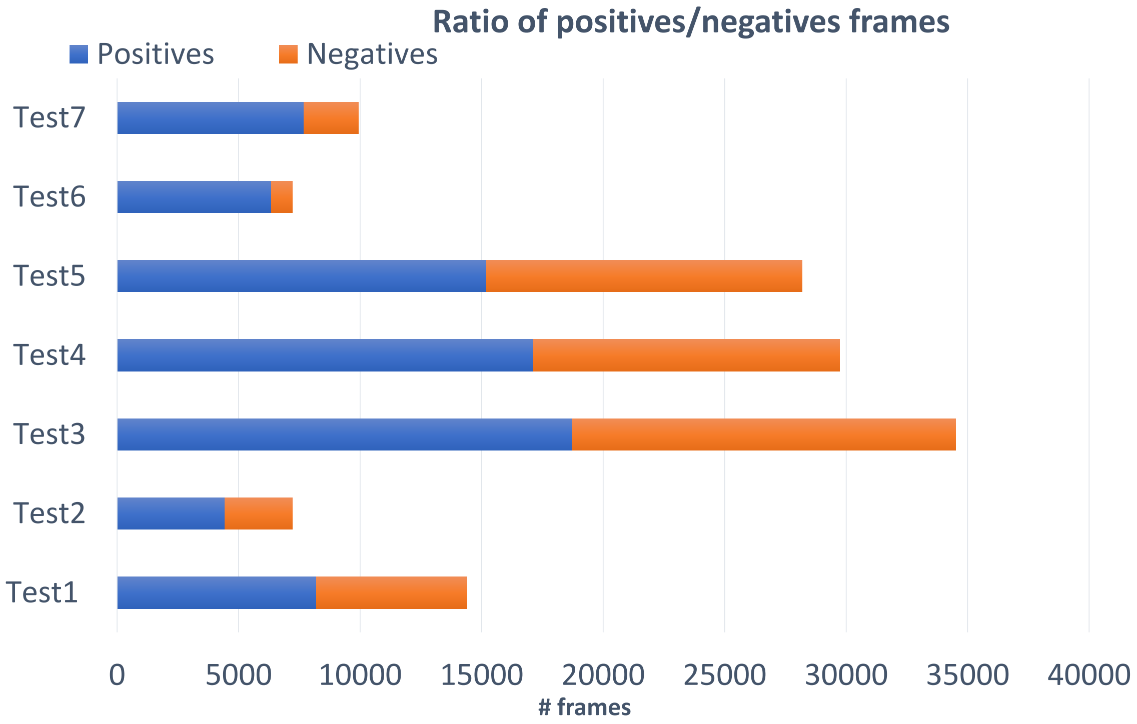


Figure 3.9: Number of “positive” frames belonging to the 57 points of interest compared to the number of “negative” frames (i.e., frames where there are not points of interest).

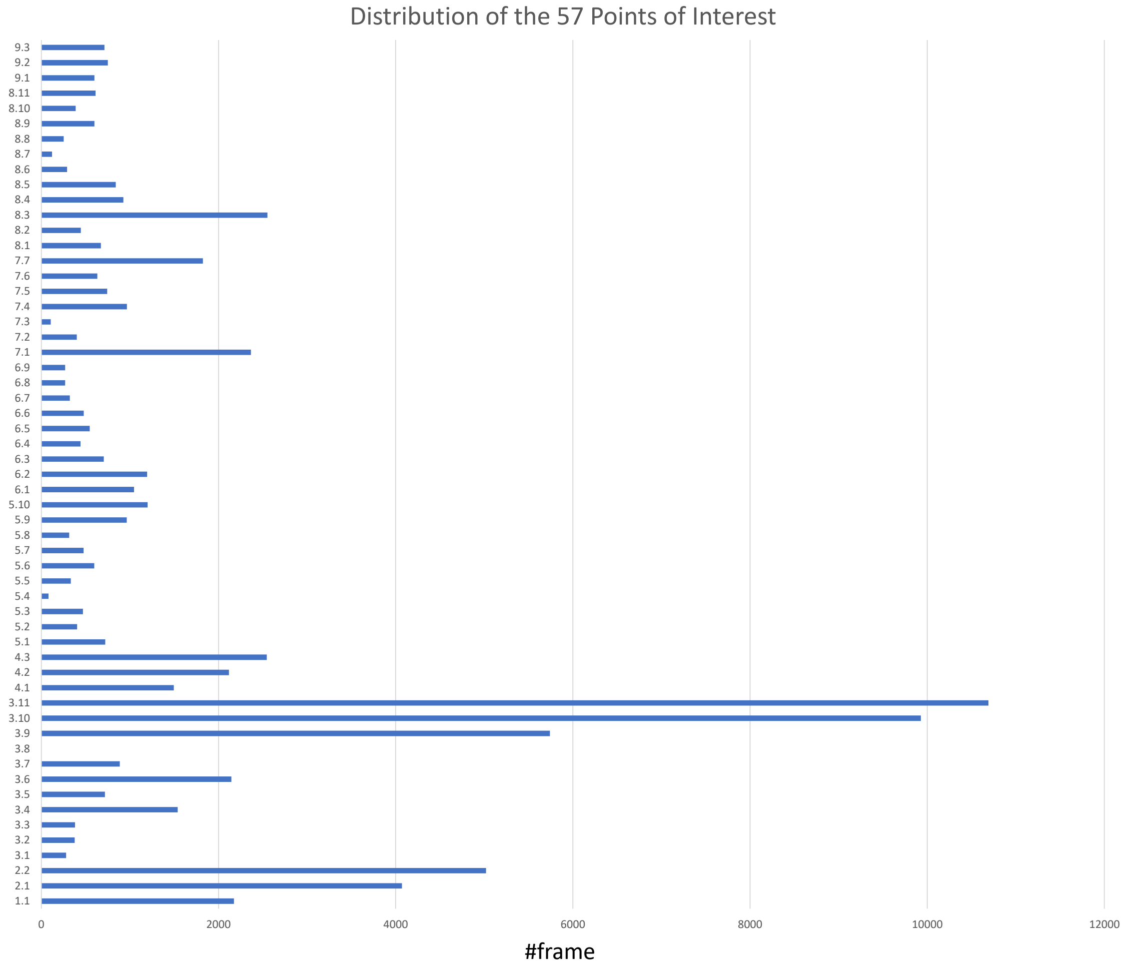


Figure 3.10: Labels distribution related to the 57 points of interest belonging to the 7 test videos.

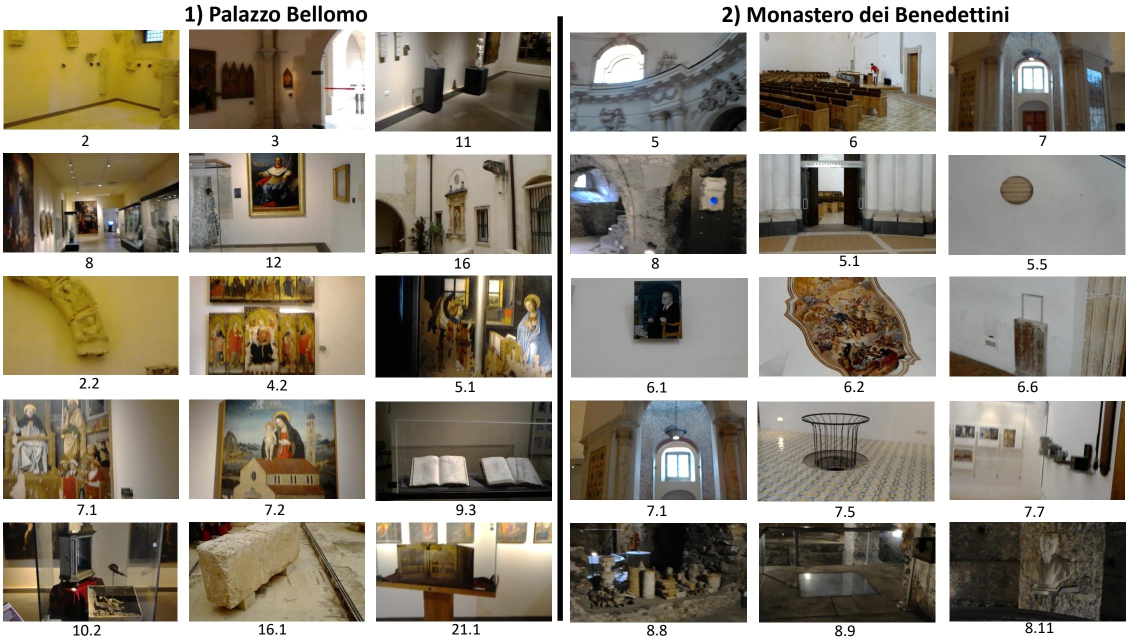


Figure 3.11: Sample frames from the two cultural sites belonging to EGO-CH: 1) Palazzo Bellomo, 2) Monastero dei Benedettini. The first two rows show frames extracted from the training videos and related to the environments, whereas the remaining rows show frames of the training videos related to POIs.

significantly the UNICT-VEDI dataset [144] described in Section 3.2, introducing 60 new labelled videos collected by real visitors. Specifically, the overall dataset presented in this section contains +1600 minutes of video, data from +70 more subjects, +91369 bounding box annotations and an additional cultural site "Palazzo Bellomo" comprising 22 environments and 191 points of interest. We included only the Training and Validation videos of the UNICT-VEDI dataset belonging to the 4 considered environments.

3.3.1 Data Collection

The dataset has been acquired using a head-mounted Microsoft HoloLens device[6] in two cultural sites located in Sicily, Italy: 1) Palazzo Bellomo (Table 3.11), located in Siracusa, and 2) Monastero dei Benedettini Table 3.12), located in Catania.

[6] https://www.microsoft.com/it-it/hololens

Table 3.11: Details regarding the cultural site ”Palazzo Bellomo”.

Subset	Resolution	FPS	AVG Time (min)	# POIs	#environments	bbox annotations	temporal segments
Training	1280x720	29.97	1.4	191	22	56686	57
Test	1280x720	29.97	31.27	191	22	13402	340

Table 3.12: Details regarding the cultural site ”Monastero dei Benedettini”.

Subset	Resolution	FPS	AVG Time (min)	# POIs	#environments	bbox annotations	temporal segments
Training	1216x684	24.00	2.2	35	4	33366	48
Validation	1216x684	24.00	3.5	35	4	2235	20
Test	1408x792	30.03	21	35	4	71310	455

Palazzo Bellomo

This cultural site is composed of 22 environments (see the map in Figure 3.12) and contains 191 Points of Interest (e.g., statues, paintings, etc.). Figure 3.13 and Figure 3.14 report some frames related to the different environments and some points of interest.

To acquire training videos we followed the same acquisition protocol used to acquire the UNICT-VEDI dataset (see Section 3.2. In the case of outdoor environments (e.g., courtyards), we collected multiple videos to include different lighting conditions. We have collected a total of 57 training video in this cultural site. We collected at least one training video for each of the 22 environments and at least a training video for each of the 26 points of interest. These 26 points of interest have been suggested by the site manager as main points of interest. We acquired a total of 48 training videos for this cultural site containing 22 environments and 191 points of interest (we labeled all the points of interest that appears in the video, not just the suggested one). Figure 3.13 shows some frames acquired in the considered cultural site, whereas Figure 3.15 reports the number/percentage of frames acquired in each environment. Table 3.13 details the list of the acquired training videos. Some of the videos are related to the 22 rooms of the cultural site, whereas other are related to specific points of interest. For each video, we report its total duration, the amount of required storage, the number of frames, as well as the percentage of frames with respect to the whole training set.

Ten test videos have been collected separately asking 10 volunteers to visit the cultural site. One of the 10 videos (i.e., “Test 3”) was selected randomly and used as validation set, whereas the remaining 9 videos have been used for evaluation purposes. No specific instructions on where to go, what to look at and how much time to spend in a specific environment/POI has been provided to the visitors. Most

Table 3.13: List of training videos of “Palazzo Bellomo”.

Name	Time (s)	Storage (MB)	#frame	%frame
1.0_Sala1	124	156.229	3721	3,13%
2.0_Sala2	117	148.480	3525	2,96%
3.0_Sala3	100	125.924	3000	2,52%
3.0_Sala3_S	73	92.589	2200	1,85%
4.0_Sala4	97	122.941	2914	2,45%
5.0_Sala5	99	126.213	2992	2,51%
6.0_Sala6	87	110.451	2630	2,21%
7.0_Sala7	113	143.257	3402	2,86%
8.0_Sala8	147	186.470	4427	3,72%
9.0_Sala9	143	180.971	4298	3,61%
10.0_Sala10	71	90.697	2154	1,81%
11.0_Sala11	104	131.983	3145	2,64%
12.0_Sala12	82	103.785	2463	2,07%
13.0_Sala13	101	128.013	3040	2,55%
14.0_CortiledegliStemmi	104	131.962	3131	2,63%
14.0_CortiledegliStemmi_S	90	113.822	2722	2,29%
15.0_SalaCarrozze	108	136.968	3259	3,12%
16.0_CortileParisio	124	156.605	3718	3,12%
16.0_Cortile_Parisio_S	68	86.646	2062	1,73%
17.0_Biglietteria	83	104.532	2489	2,09%
17.0_Biglietteria_S	53	68.071	1610	1,35%
18.0_Portico	126	159.800	3791	3,18%
18.0_Portico_S	63	80.044	1910	1,60%
19.0_ScalaCatalana	97	123.010	2918	2,45%
19.0_ScalaCatalana_S	116	110.063	3481	2,92%
20.0_Loggetta	80	101.584	2425	2,04%
20.0_Loggetta_S	58	73.722	1744	1,46%
21.0_BoxSala8	85	107.540	2562	2,15%
22.0_AreaSosta	64	81.934	1945	1,63%
22.0_Area_Sosta_S	52	65.340	1560	1,31%
2.1_Sala2_Acquasantiera	54	68.445	1623	2,94%
2.2_Sala2_FrammentiArchitett.	46	58.265	1393	2,53%
2.3_Sala2_LastraconLeoni	47	60.199	1427	2,59%
3.1_Sala3_MadonnainTrono	65	83.198	1972	3,58%
3.2_Sala3_FrammentoS.Leonardo	37	47.061	1113	2,02%
4.1_Sala4_MadonnainTrono	75	94.767	2252	4,08%
4.2_Sala4_MonumentoE.d’Aragona	86	108.296	2580	4,68%
4.3_Sala4_TrasfigurazioneCristo	76	96.106	2277	4,13%
4.4_Sala4_Piatti	49	62.281	1474	2,67%
5.1_Sala5_Annunciazione	76	96.952	2295	4,16%
5.2_Sala5_LibroD’OreMiniato	46	59.011	1406	2,55%
5.3_Sala5_LastraG.Cabastida	100	127.023	3017	5,47%
5.4_Sala5_MadonnadelCardillo	61	77.568	1829	3,32%
7.1_Sala7_DisputaS.Tommaso	74	94.188	2234	4,05%
7.2_Sala7_TraslazioneSantaCasa	76	96.045	2281	4,14%
7.3_Sala7_MadonnacolBambino	90	113.202	2696	4,89%
8.1_Sala8_ImmacolataConcezione	82	104.570	2483	4,50%
9.1_Sala9_AdorazionedeiMagi	60	76.171	1803	3,27%
9.2_Sala9_S.ElenaCostantinoeMadonna	76	96.227	2283	4,14%
9.3_Sala9_TaccuinidiDisegni	70	89.647	2121	3,85%
10.1_Sala10_MartirioS.Lucia	58	74.248	1759	3,19%
10.2_Sala10_VoltodiCristo	64	80.896	1917	3,48%
11.1_Sala11_MiracolodiS.Orsola	66	84.297	2002	3,63%
11.2_Sala11_Immacolata	73	92.424	2196	3,98%
16.1_CortileParisio_LapidiEbraiche	85	108.098	2563	4,65%
16.1_CortileParisio_LapidiEbraiche_S	67	85.173	2031	3,68%
21.1_BoxSala8_StoriedellaGenesi	70	88.300	2099	3,81%
AVG	**81.72**	**103.02**	**2462.53**	**2.07%**

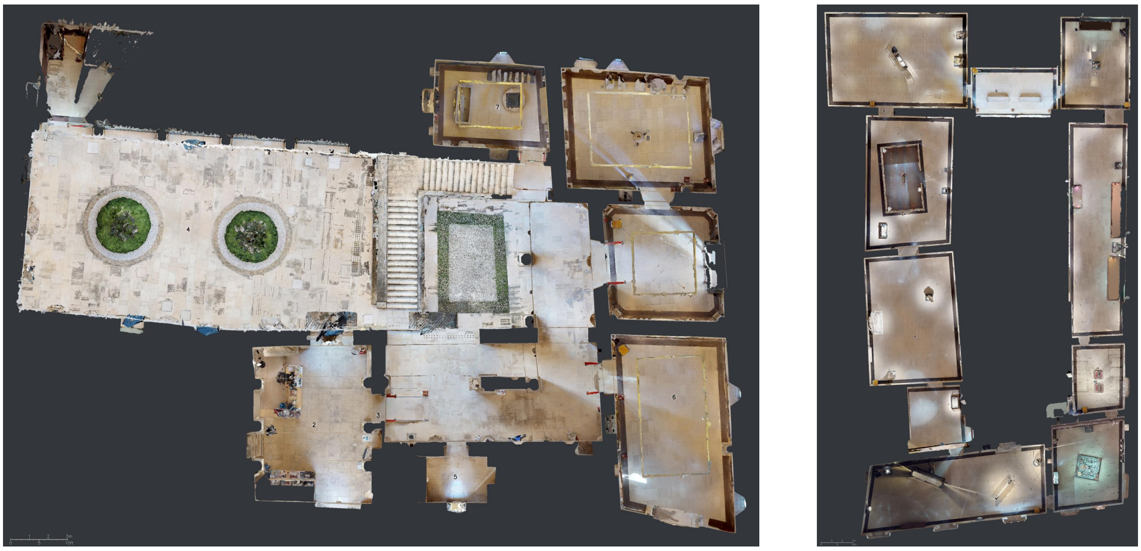


Figure 3.12: The map of the cultural site "Palazzo Bellomo". The base floor is on the left, while the first floor is on the right.

of the subjects had limited confidence with the cultural site. This provided a natural means to collect realistic data of visitors exploring the environments and observing Points of Interest. Table 3.14 reports the list of the 10 test videos acquired by volunteers visiting the cultural site. For each video, we report its total duration, the amount of required storage, the number of frames, the number of environments encountered in the video, as well as the sequence of environments, as visited by the subject acquiring the video. All the videos have a resolution of 1280 × 720 pixels and a frame-rate of 29.97 fps. The average duration of test videos is 31.27 min, with the longest one being 50.23 min.

We also included 191 reference images related to the considered POIs to be used for one-shot image retrieval task. The images are akin to the images generally included in museum catalogs. Figure 3.16 shows some examples of such reference images.

Monastero dei Benedettini

This cultural site is composed of 4 environments and contains 35 Points Of Interest (see the map in Figure 3.17). Figure 3.18 and Figure 3.19 report some frames related to the 4 different environments and some of the points of interest. Table 3.15 reports details on the acquired training videos, highlighting the total duration of the videos,

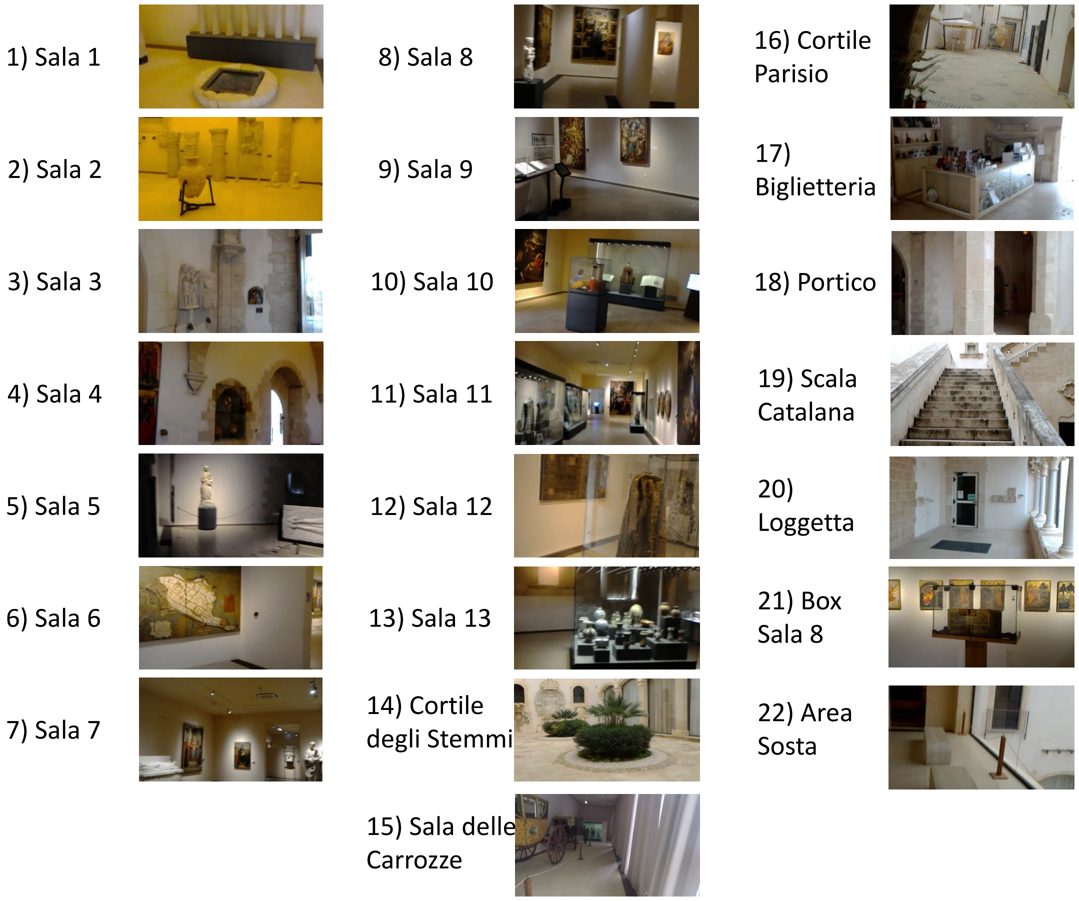


Figure 3.13: Sample frames for each of the 22 considered environments of "Palazzo Bellomo".

Figure 3.14: Sample frames of points of interest of "Palazzo Bellomo".

Environment	#video	#frame
1 Sala1	1	3721
2 Sala2	4	7968
3 Sala3	4	8285
4 Sala4	5	11497
5 Sala5	5	11461
6 Sala6	1	2630
7 Sala7	4	10613
8 Sala8	2	6910
9 Sala9	4	10505
10 Sala10	3	5830
11 Sala11	3	7343
12 Sala12	1	2463
13 Sala13	1	3040
14 Cortile degli Stemmi	2	5853
15 Sala delle Carrozze	1	3259
16 Cortile Parisio	4	10374
17 Biglietteria	2	4099
18 Portico	2	5701
19 Scala Catalana	2	6399
20 Loggetta	2	4169
21 Box Sala8	2	4661
22 Area Sosta	2	3505
Total	**57**	**140286**

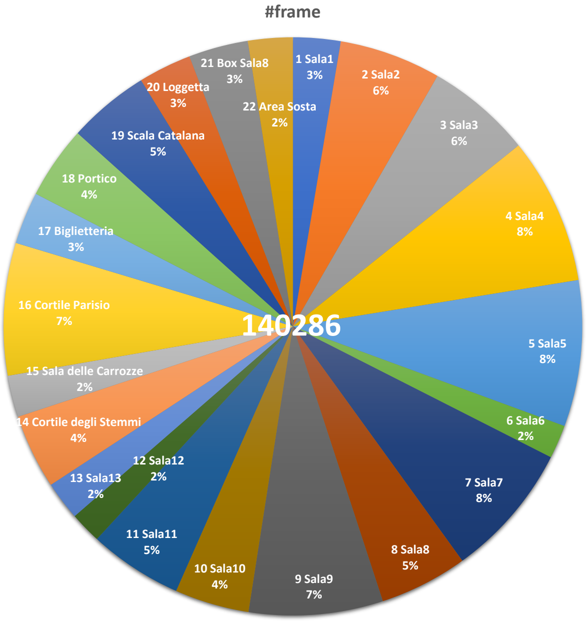


Figure 3.15: Number of training videos collected in each environment and corresponding number of frames for the cultural site "Palazzo Bellomo" (left), along with a pie chart representation of the same data (right).

Table 3.14: List of test videos of "Palazzo Bellomo".

Name	Time (s)	MB	#Frame	%Frame	#Environments	Environments - Temporal sequence
Test1	1906	2.400.360	57123	11,13%	22	16->17->18->1->18->3->2->3->18->4->18->14->15->14->19->20->6->5->6->7->21->8->9->22->10->11->12->13->5->6->20->19->18->17
Test2	1413	1.435.096	42348	8,25%	22	16->17->18->1->18->3->2->3->18->4->18->14->15->14->19->20->6->5->6->7->8->21->8->9->22->10->11->12->13->5->6->20->19->18->17
Test3	1830	2.304.410	54845	10,69%	22	16->17->18->1->18->3->2->3->18->4->18->14->15->14->19->20->6->5->13->12->11->10->22->9->8->21->8->7->6->20->19->18->17
Test4	1542	1.942.200	46214	5,49%	22	16->17->18->1->18->3->2->3->18->4->18->14->15->14->19->20->6->5->6->7->8->21->8->9->22->10->11->12->13->5->6->20->19->18->17
Test5	1034	1.302.612	30989	9,00%	22	16->17->18->1->18->3->2->3->18->4->18->14->15->14->19->20->6->5->6->7->8->9->22->10->11->12->13->5->6->20->19->18->17
Test6	1949	2.273.926	58411	11,38%	22	16->17->8->1->18->3->2->3->18->14->15->14->19->20->6->5->13->5->6->7->8->21->8->9->22->10->11->12->11->10->22->9->8->7->6->20->19->18->17
Test7	1332	1.677.047	39920	7,78%	22	16->17->14->15->14->18->1->18->3->2->3->18->4->18->19->20->6->5->6->7->8->21->8->9->22->10->11->12->13->5->6->20->19->18->17->16
Test8	3023	3.806.383	90599	17,65%	22	16->17->14->5->14->18->4->18->3->2->3->18->1->18->19->20->6->5->13->12->11->10->22->9->8->21->8->7->6->20->19->14
Test9	2236	2.815.878	67013	13,05%	22	16->17->18->1->18->3->2->3->18->4->18->14->15->14->19->20->6->5->13->12->11->10->22->9->8->21->8->7->6->20->19->14->17
Test10	858	1.080.389	25714	5,01%	22	16->17->14->19->20->6->7->8->21->8->9->22->10->11->12->13->5->6->20->19->18->4->18->1->18->3->2->3->18

Figure 3.16: Sample references images related to the cultural site "Palazzo Bellomo".

the required storage, the number of frames and the percentage of frames with respect to the whole training set.

Differently from "Palazzo Bellomo", the POIs belonging to this cultural site include both objects such as paintings and statues as well as architectural elements, such as pavements, which cannot be easily recognized using object detection techniques as noted in [145]. See Figure 4.7(right) for some qualitative examples of the considered objects. Training videos have been collected with the same acquisition modality considered for the "Palazzo Bellomo" cultural site. Figure 3.20 reports the number/percentage of frames acquired in each environment.

Five validation videos have been collected by asking volunteers to visit the cultural site following the same protocol used for "Palazzo Bellomo". Training and validation videos have a resolution of 1216 × 684 pixels and a frame-rate of 24 fps. Table 3.16 shows the number of frames belonging to each video (left) and the number of frames belonging for each class (right).

Additionally, we collected 60 test videos by asking real visitors inexperienced with both the research project and its goals and the HoloLens device to freely visit the cultural site. No specific instructions have been given to the visitors, who were free to explore the 4 environments and the 35 POIs. This allowed us to obtain realistic

Table 3.15: List of training videos of “Monastero dei Benedettini”

Name	**Time (s)**	**Storage (MB)**	**#frame**	**%frame**
5.0_Antirefettorio	247	244.500	5933	1,21%
5.0_Antirefettorio1	263	260.395	6315	1,29%
6.0_SantoMazzarino	241	239.007	5800	1,18%
7.0_Cucina	239	237.268	5753	1,17%
8.0_Ventre	679	832.844	20385	4,15%
5.1_Antirefettorio_PortaA.S.Mazz.Ap.	67	67.001	1630	0,33%
5.1_Antirefettorio_PortaA.S.Mazz.Ch.	74	73.589	1785	0,36%
5.2_Antirefettorio_PortaMuseoFab.Ap.	50	49.840	1211	0,25%
5.2_Antirefettorio_PortaMuseoFab.Ch.	62	61.846	1503	0,31%
5.3_Antirefettorio_PortaAntiref.	51	58.557	1537	0,31%
5.4_Antirefettorio_PortaRef.Piccolo	54	53.767	1306	0,27%
5.5_Antirefettorio_Cupola	57	56.089	1377	0,28%
5.6_Antirefettorio_AperturaPavimento	55	54.586	1322	0,27%
5.7_Antirefettorio_S.Agata	48	47.820	1165	0,24%
5.8_Antirefettorio_S.Scolastica	58	57.919	1407	0,29%
5.9_Antirefettorio_ArcoconFirma	62	76.700	1864	0,38%
5.10_Antirefettorio_BustoVaccarini	65	64.298	1563	0,32%
6.1_SantoMazzarino_QuadroS.Mazz.	71	70.401	1716	0,35%
6.2_SantoMazzarino_Affresco	213	211.424	5124	1,04%
6.3_SantoMazzarino_PavimentoOr.	99	98.691	2397	0,49%
6.4_SantoMazzarino_PavimentoRes.	69	69.148	1675	0,34%
6.5_SantoMazzarino_BassorilieviManc.	117	115.348	2823	0,57%
6.6_SantoMazzarino_LavamaniSx	151	149.882	3637	0,74%
6.7_SantoMazzarino_LavamaniDx	93	92.928	2256	0,46%
6.8_SantoMazzarino_TavoloRelatori	150	148.661	3603	0,73%
6.9_SantoMazzarino_Poltrone	108	107.374	2604	0,53%
7.1_Cucina_Edicola	369	437.219	11086	2,26%
7.2_Cucina_PavimentoA	52	52.163	1268	0,26%
7.3_Cucina_PavimentoB	52	52.244	1266	0,26%
7.4_Cucina_PassavivandePavim.Orig.	81	80.733	1961	0,40%
7.5_Cucina_AperturaPav.1	57	57.170	1385	0,28%
7.5_Cucina_AperturaPav.2	53	52.875	1280	0,26%
7.5_Cucina_AperturaPav.3	62	62.320	1509	0,31%
7.6_Cucina_Scala	77	76.587	1856	0,38%
7.7_Cucina_SalaMetereologica	156	154.394	3748	0,76%
8.1_Ventre_Doccione	103	102.683	2492	0,51%
8.2_Ventre_VanoRacc.Cenere	126	124.837	3026	0,62%
8.3_Ventre_SalaRossa	300	296.792	7202	1,47%
8.4_Ventre_ScalaCucina	214	212.372	5152	1,05%
8.5_Ventre_CucinaProvv.	148	146.379	3553	0,72%
8.6_Ventre_Ghiacciaia	69	68.562	1668	0,34%
8.6_Ventre_Ghiacciaia1	266	263.817	6398	1,30%
8.7_Ventre_Latrina	102	100.542	2468	0,50%
8.8_Ventre_OssaScarti	154	152.861	3713	0,76%
8.8_Ventre_OssaScarti1	36	36.345	886	0,18%
8.9_Ventre_Pozzo	300	297.167	7209	1,47%
8.10_Ventre_Cisterna	57	56.572	1384	0,28%
8.11_Ventre_BustoP.Tacchini	110	109.520	2662	0,54%
AVG	**133.06**	**137.38**	**3351.31**	**1.04%**

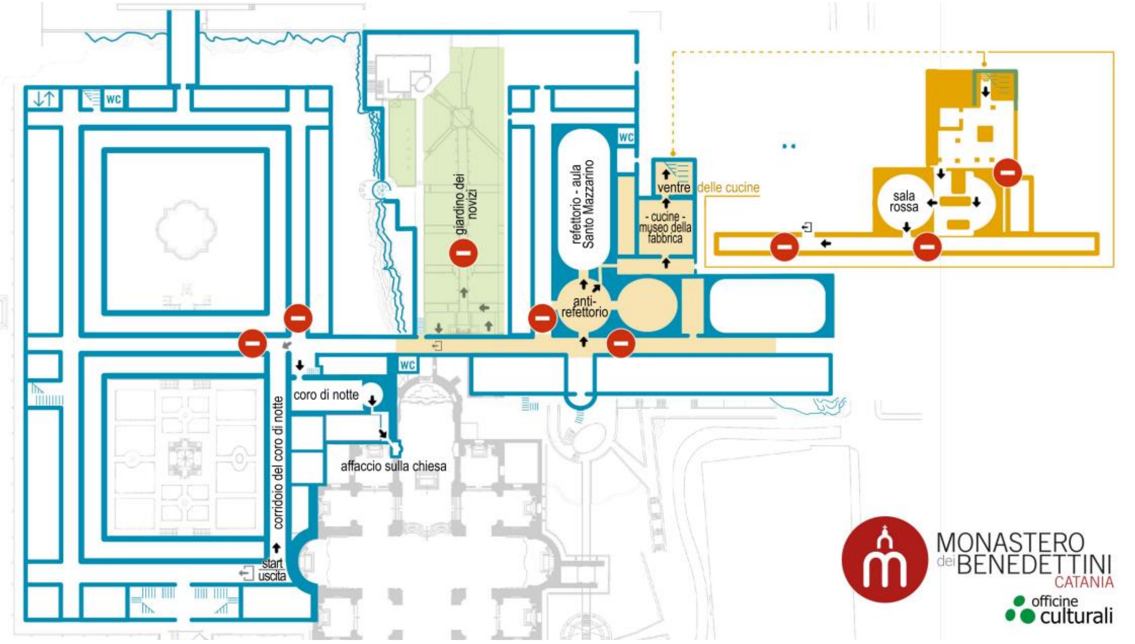
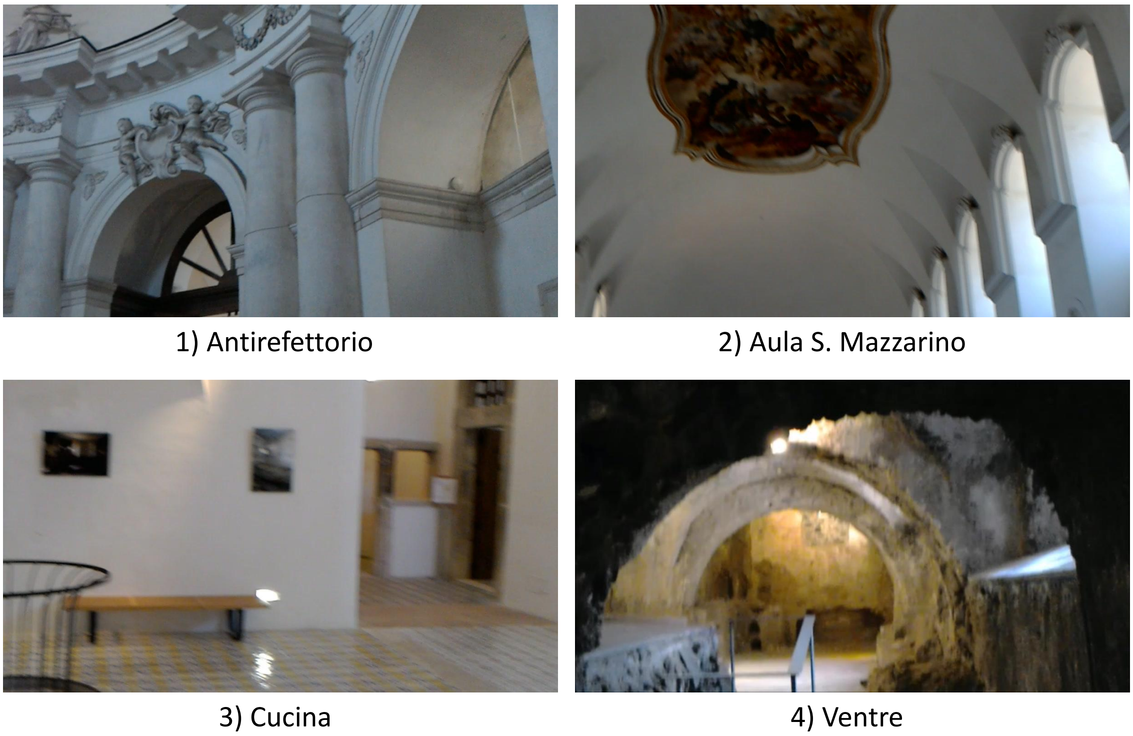


Figure 3.17: The map of the "Monastero dei Benedettini" cultural site. The rows indicate the path which the visitors followed during the acquisitions.

data of how a visitor would move in a cultural site. Test videos have been collected over a period of three months. Moreover, at the end of the visit, we administered the visitor a survey, the content of which is described in Section 3.3.2. The 60 test videos have a resolution of 1408 × 792 pixels and a frame-rate of 30.03 fps. The average video length is 21 min, with the maximum length being 42 min. Similarly to "Palazzo Bellomo", we include 35 reference images related to the considered POIs for one-shot image retrieval task. Figure 3.21 shows some example of reference images.

3.3.2 Annotations

We labeled the EGO-CH dataset with temporal annotations, bounding box annotations around the points of interest observed by the visitors and we also collected surveys associated to the visits related to the "Monastero dei Benedettini" cultural site.



1) Antirefettorio

2) Aula S. Mazzarino

3) Cucina

4) Ventre

Figure 3.18: Sample frames from the 4 considered environments of "Monastero dei Benedettini".

Temporal Labels

All test and validation videos have been temporally labeled to indicate in every frame the environment in which the visitor is located and the observed point of interest, if any. If the visitor is not located in one of the considered environment (e.g., a stair), the frame is marked as "negative". Examples of "negative" frames are reported in Figure 3.22.

It is worth noting that there are no negative frames in "Palazzo Bellomo" since all environments are part of the museum, whereas negative frames are contained in "Monastero dei Benedettini". This is due to the different nature of the two sites: "Palazzo Bellomo" is a museum, consisting in a limited set of rooms, whereas "Monastero dei Benedettini" is a much more complex environment including many corridors and stairs which have not been labeled as locations of interest for visitors. Similarly, we mark as "negative" all frames in which the visitor is not observing any of the considered POIs. Each location is identified by a number that denotes

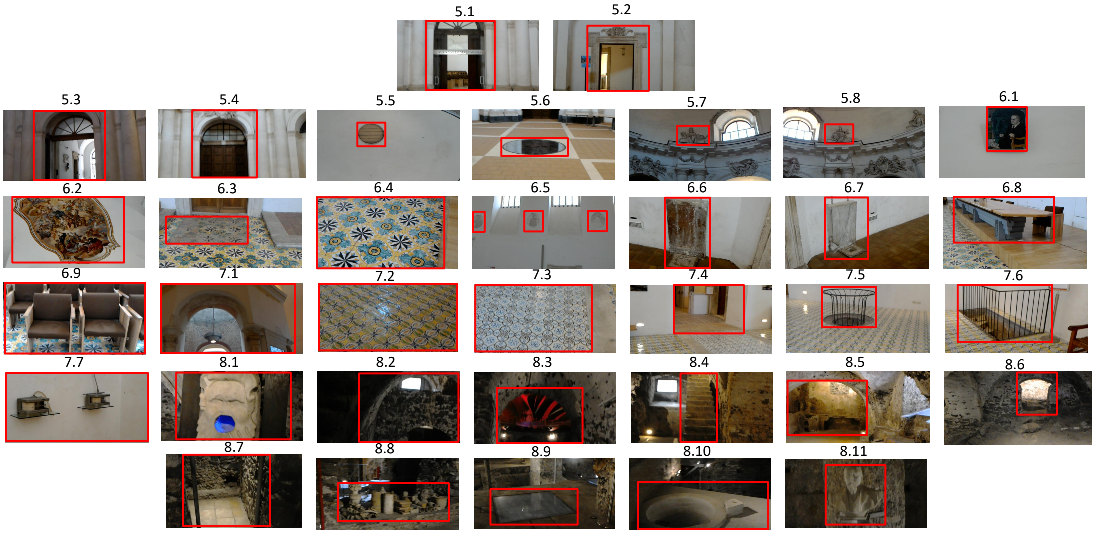


Figure 3.19: Sample frames from the 35 considered POIs of “Monastero dei Benedettini”, with the related bounding box annotations.

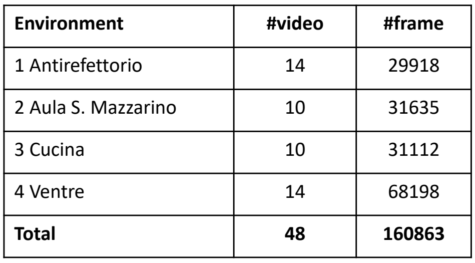
Table 3.16: List of validation videos of “Monastero dei Benedettini”.

Name	#frame	Class	#frame
Test1	4141	1 Antirefettorio	88613
Test3	18678	2 Aula S. Mazzarino	8395
Test4	13731	3 Cucina	9712
Test5	15958	4 Ventre	20513
Test7	1124	Negatives	6399
Total	**53632**	**Total**	**53632**

a specific environment ($1-22$ for “Palazzo Bellomo” and $1-4$ for “Monastero dei Benedettini”). Each point of interest is denoted by a code in the form X.Y (e.g., 3.5) where “X” denotes the environment in which the point of interest is located and “Y” identifies the point of interest. See Figure 4.7 for some examples.

Bounding Box Annotations

A subset of frames from the whole dataset (sampled at 1 fps) has been labeled with bounding boxes indicating the presence and locations of all POIs. Specifically, each POI has been labeled with a tuple $(class, x, y, w, h)$ indicating the class of the POI



Environment	#video	#frame
1 Antirefettorio	14	29918
2 Aula S. Mazzarino	10	31635
3 Cucina	10	31112
4 Ventre	14	68198
Total	**48**	**160863**

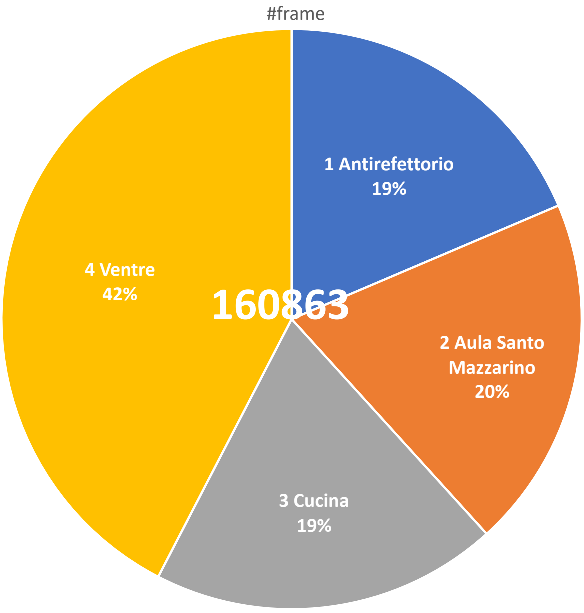


Figure 3.20: Number of training videos collected in each environment and corresponding number of frames for the cultural site “Monastero dei Benedettini” (left), along with a pie chart representation of the same data (right).

and its bounding box information. It is worth mentioning that, as noted in [145], a POI can be an object (e.g., a painting or a statue) or a different element (e.g., a pavement or a specific location), which cannot be strictly defined as an object. Figure 3.19 shows some example of labeled frames from the training set of the “Monastero dei Benedettini”. Indeed, the kind of POIs contained in a cultural site depends on the nature of the site itself. In EGO-CH, “Palazzo Bellomo” contains only objects as POIs, whereas “Monastero dei Benedettini” contains both objects and other elements. Nevertheless, all elements are labeled with class type and bounding box annotations. Figure 3.23 shows examples of labeled frames from the 60 visits of “Monastero dei Benedettini”.

Surveys

The 60 test videos collected in the “Monastero dei Benedettini” are associated with surveys which have been administered to the visitors at the end of the visits. Specifically, the visitors are asked to rate a subset of 33 out of the 35 Points Of Interest (a picture of each point is shown) or specify if any of them had not been seen it during

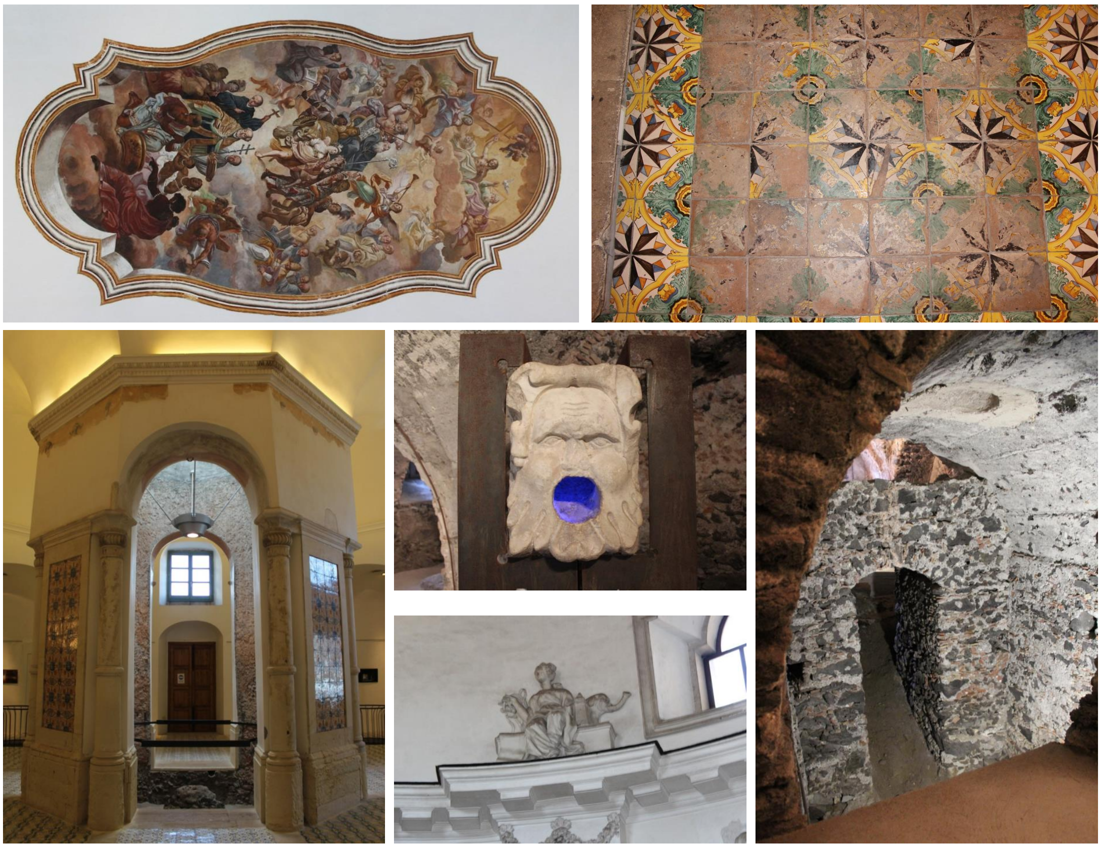


Figure 3.21: Sample references images related to the “Monastero dei Benedettini”.

the visit. The rating is expressed as a number ranging from −7 (not liked) to +7 (liked). To check that answers are provided correctly, we also introduced distractors in the form of POIs not present in the cultural site. Figure 3.24 reports an example of the questions asked to the visitor through the survey.

3.3.3 EGO-CH for Semantic Segmentation

Detecting the object of interest at the bounding box level can be limited in the context of cultural heritage (because the shape and size of artworks has high variability). For this reason we also focused on the task of semantic object segmentation at the pixel-level in cultural sites. We extended the EGO-CH dataset [147] adding

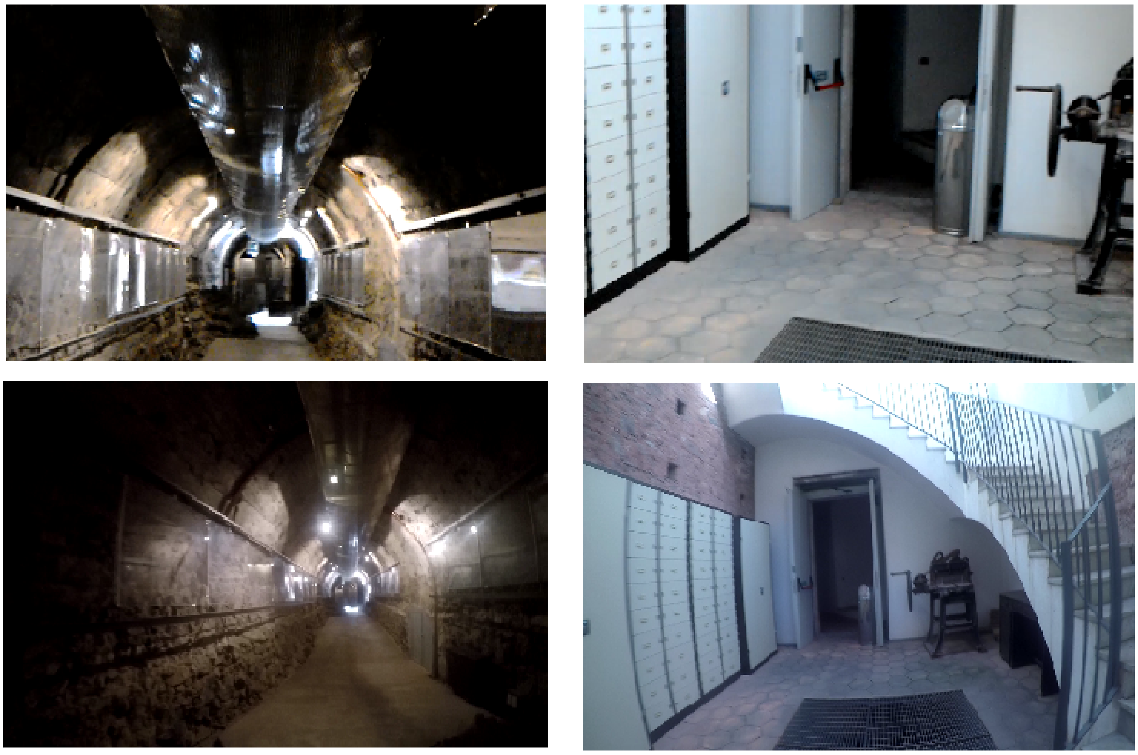


Figure 3.22: Sample frames from "Monastero dei Benedettini" marked as "negative locations".

Figure 3.23: Some example bounding box annotations from the cultural site "Monastero dei Benedettini".

real and synthetic images depicting 24 artworks located in 11 different environments of the Galleria Regionale Palazzo Bellomo[7]. All images are paired with semantic segmentation masks indicating the presence of artworks at pixel-level. The EGO-CH-OBJ-SEG dataset is publicly available at: https://iplab.dmi.unict.it/EGO-CH-OBJ-SEG/.

Real Images

We consider real images depicting artworks belonging to the EGO-CH dataset [146]. We concentrate on the subset of the data acquired in the Galleria Regionale di Palazzo Bellomo cultural site and select 24 artworks for our study. Since images of

[7] http://www.regione.sicilia.it/beniculturali/palazzobellomo/

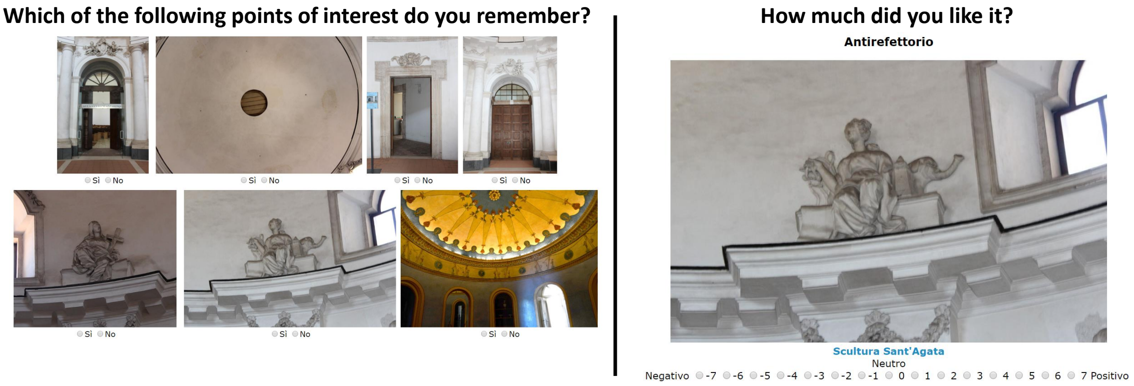

Which of the following points of interest do you remember?
○ Sì ○ No
○ Sì ○ No
○ Sì ○ No
○ Sì ○ No
○ Sì ○ No
○ Sì ○ No
○ Sì ○ No
How much did you like it?
Antirefettorio
Scultura Sant'Agata
Neutro
Negativo ○ -7 ○ -6 ○ -5 ○ -4 ○ -3 ○ -2 ○ -1 ○ 0 ○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7 Positivo

Figure 3.24: An illustration of the collection of surveys from the visitors of the cultural site.

Table 3.17: Details about the proposed dataset, including the number of real and synthetic training, validation and test images.

	Resolution	#Artworks	#Environments	Segmentation Masks	Training Images	Val. Images	Test Images	All Images
Real	1280x720	24	11	56241	4740 (85%)	170 (3%)	678 (12%)	5580
Synthetic	1280x720	24	11	24000	12000 (50%)	1200 (5%)	10800 (45%)	24000

The number of segmentation masks is greater than the number of images due to the fact that some images have multiple annotations.

EGO-CH are annotated only with bounding boxes, we have manually labelled 5588 images. Specifically we annotated 4740 images from the training set of [146] and 848 images from its test set. Moreover, 170 images of the 848 images are used for validation, whereas the remaining 678 are used for test. We used the VGG annotation tool [148] to obtain all annotations. Table 3.17 reports some details about the dataset and summarizes the number of training and test images belonging to the dataset, whereas Figure 4.7 reports examples of real images for each of the 24 artworks together with the associated segmentation masks. Class labels and the related number of manually annotated segmentation masks are reported in Table 3.18.

Synthetic Images

To automatically obtain a large number of synthetic images with the related semantic mask annotations, we developed a tool based on Blender [149]. Given a 3D model of a cultural site, the tool allows to manually label the artworks in the 3D coordinate system. It then automatically generates RGB images of the artworks acquired from multiple points of view, together with the related segmentation masks. To generate

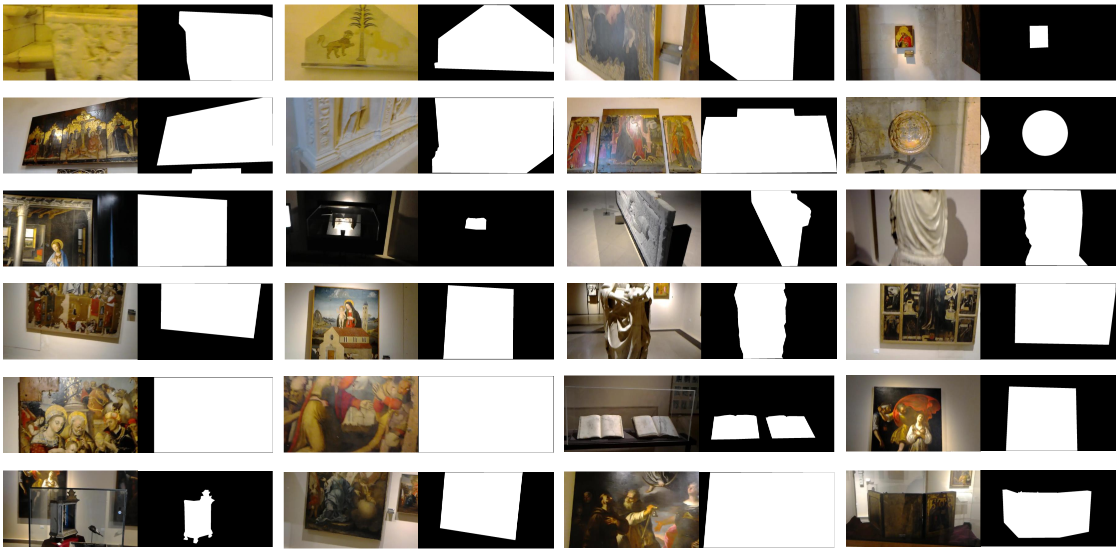


Figure 3.25: Examples of real images depicting the 24 artworks along with the annotated segmentation masks.

Table 3.18: Classes related to the dataset with number of annotation masks. Each point of interest is denoted by an ID in the form X.Y (e.g.,2.1) where “X” denotes the environment in which the artwork is located and “Y” identifies the artwork.

ID	Class	Annotations	ID	Class	Annotations	ID	Class	Annotations
2.1	Acquasantiera	244	5.1	Annunciazione	303	9.1	AdorazionedeiMagi	230
2.3	LastraconLeoni	248	5.2	LibroD’OreMin.	253	9.2	S.ElenaCost.eMadon.	247
3.1	MadonnainTrono	237	5.3	LastraG.Cabastida	307	9.3	TaccuinidiDisegni	212
3.2	FrammentoS.Leo	186	5.4	MadonnadelCard.	223	10.1	MartirioS.Lucia	196
4.1	MadonnainTrono	245	7.1	DisputaS.Tomm.	200	10.2	VoltodiCristo	210
4.2	MonumentoE.d’Aragona	222	7.2	TraslazioneS.Casa	279	11.1	MiracolodiS.Orsola	250
4.3	Trasf.Cristo	233	7.3	MadonnacolBam.	231	11.2	Immacolata	219
4.4	Piatti	208	8.1	ImmacolataConc.	245	21.1	StoriedellaGenesi	196

the synthetic images, we used the 3D model of Palazzo Bellomo acquired in [150] (see Figure 3.26), which is the same scenario where real images have been acquired.

Formally, given a set of objects $O = [o_1, o_2, .., o_n]$ and a set of different colours $C = [c_1, c_2, .., c_m]$, for each object $o \in O$ we assign an identification color $c \in C$. The framework, after setting the number of desired images for each object, captures the RGB images of the objects varying the point of view of the camera. Then, a manual colorization phase for each object's texture with the related color is needed. Finally, the framework captures the same images with the same camera position acquired in the previous step. In this manner, we obtain for each RGB image the related semantic mask. Using the developed tool, we generated 12000 training images,

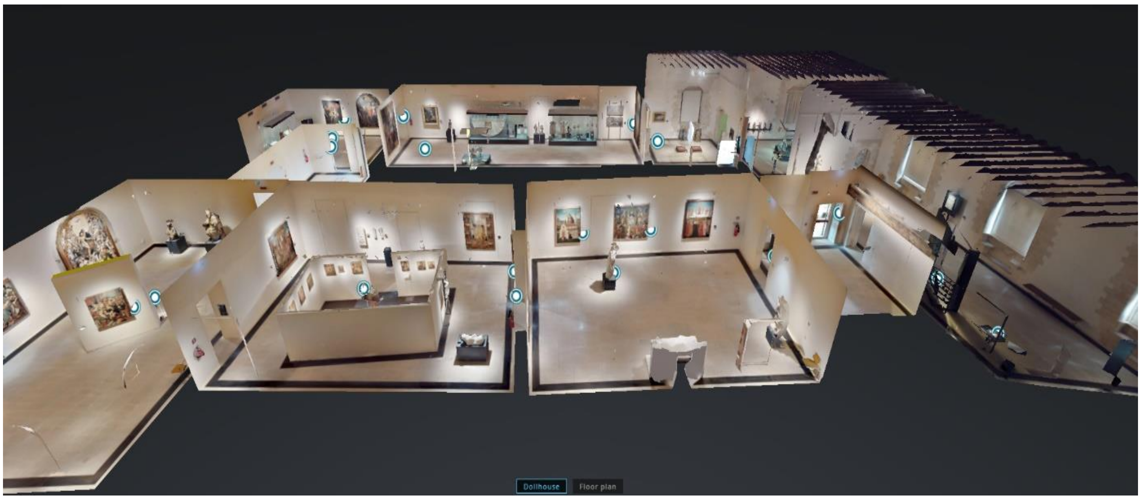


Figure 3.26: 3D model of the Galleria Regionale di Palazzo Bellomo acquired in [150].

Table 3.19: Comparative overview of the described datasets. LOC: Localization, POI-R: Points of Interest Recognition, OR: Object Retrieval, SG: Survey Generation, SOS: Semantic Object Segmentation.

Dataset	Year	Cultural Sites	Devices	Environments	Points of Interest	Tasks	Frames	Sequences	Frame with BBs	Semantic Masks	Syntethic Data?	Participants	Overlap
UNICT-VEDI	2019	1	Hololens, GoPro	9	57	LOC	508001	160	0	0	X	7	
UNICT-VEDI-POIs	2019	1	Hololens	9	57	POI-R	54248	80	54248	0	X	7	Frames from the UNICT-VEDI
EGO-CH	2020	2	Hololens	26	226	LOC, POI-R, OR, SG	3 M	180	176999	0	X	77	Training/Validation videos from the UNICT-VEDI (only 4 environtments)
EGO-CH-OBJ-SEG	2020	1	Hololens	11	24	SOS	30588	0	0	30588	\checkmark	10	Frames from the EGO-CH (only 1 cultural site)

1200 validation images and 10800 test images (see Table 3.17 for a summary). In particular, we generated 1000 images for each considered artwork. Figure 3.27 shows examples of synthetic images of the 24 artworks with the related segmentation masks.

Table 3.19 compares the described datasets reporting some details and highlighting the overlaps between them.

3.4 Room-based Localization

The room-based localization task consists in determining the room in which the visitor of a cultural site is located from egocentric images collected using a wearable device. Localization information can be used both to provide a "where am I" service to the visitor and to collect behavioral information useful for the site manager to

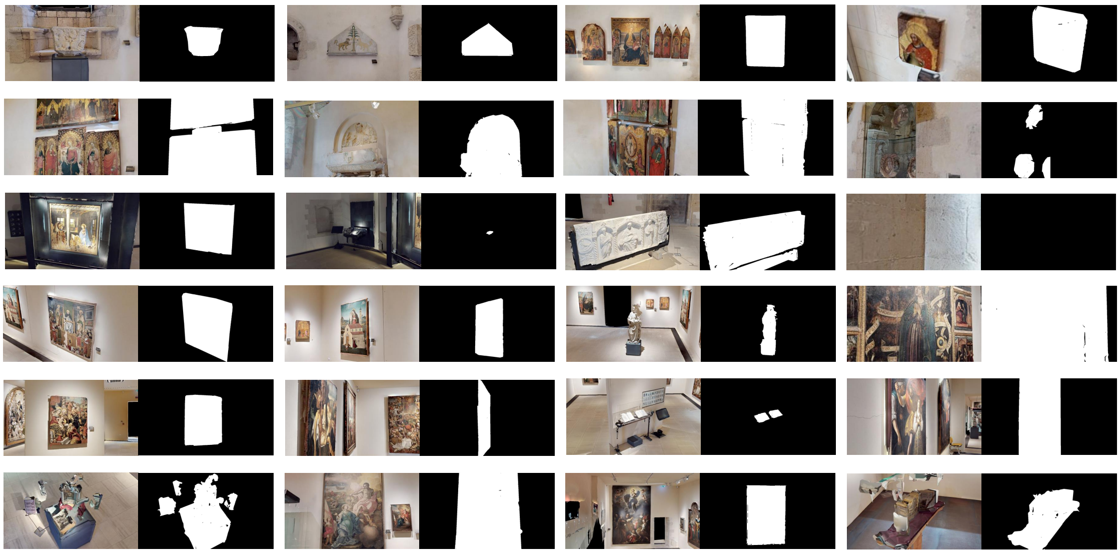


Figure 3.27: Samples of synthetic images of the 24 artworks with the related segmentation masks automatically generated using the developed tool.

understand what paths do visitors prefer and where they spend more time in the cultural site. We perform experiments and report baseline results related to the task of localizing visitors from egocentric videos on both UNICT-VEDI [144] and EGO-CH [146] datasets. To address the localization task, we consider the approach proposed in [6]. This method is particularly suited for the considered task since it can be trained with a small number of samples and includes a rejection option to determine when the visitor is not located in any of the environments considered at training time (i.e., when a frame belongs to the negative class). Moreover, as detailed in [6], the method achieves state-of-the-art results on the task of location-based temporal video segmentation, outperforming classic methods based on SVMs and local feature matching. At test time, the algorithm divides an egocentric video into temporal segments each associated to either one of the "positive" classes or, alternatively, to the "negative" class.

3.4.1 Method

At training time, we define a set of M "positive" classes. In our case, this corresponds to the set of training videos acquired for each of the considered environments.

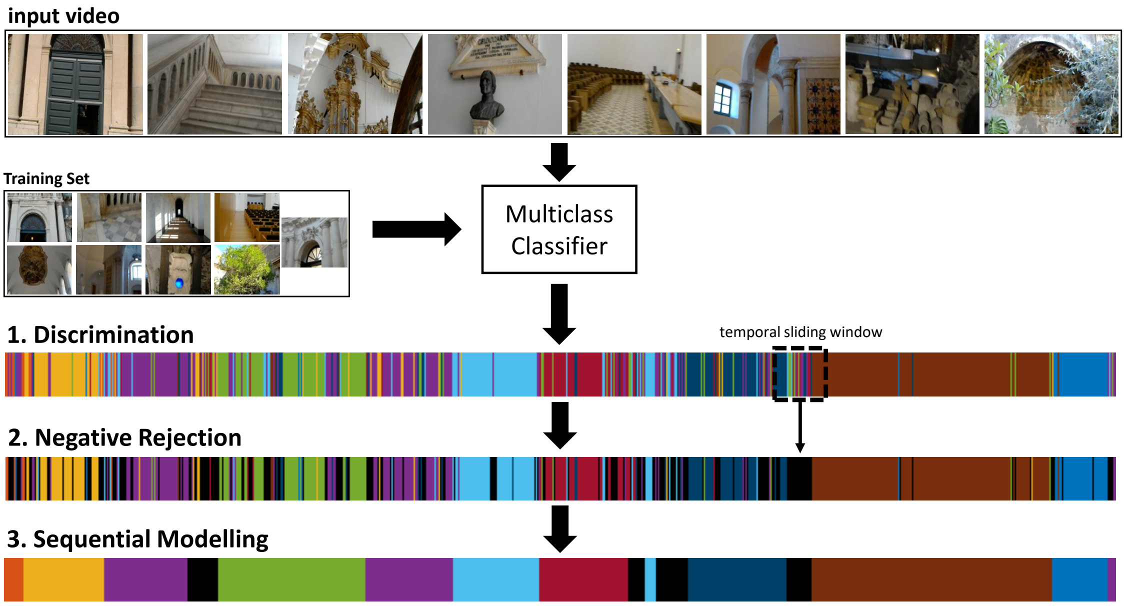


Figure 3.28: Diagram of the considered room-based localization method consisting in three steps: 1) Discrimination, 2) Negative Rejection, 3) Sequential Modeling.

At test time, an input egocentric video $\mathcal{V} = \{F_1, \ldots, F_N\}$ composed by N frames F_i is analyzed. Each frame of the video is assumed to belong to either one of the considered M positive classes or none of them. In the latter case, the frame belongs to the "negative class". Since, negative training samples are not assumed at training time due to the hugeness of data necessary to represent everything that does not belong to the chosen locations, the algorithm has to detect which frames do not belong to any of the positive classes and reject them. The goal of the system is to divide the video into temporal segments, i.e., to produce a set of P video segments $\mathcal{S} = \{s_i\}_{1 \leq i \leq P}$, each associated to a class (i.e., the room-level location). In particular, each segment is defined as $s_i = \{s_i^s, s_i^e, s_i^c\}$, where s_i^s represents the staring frame of the segment, s_i^e represents the ending frame of the segments and $s_i^c \in \{0, \ldots, M\}$ represents the class of the segment ($s_i^c = 0$ is the "negative class", while $s_i^c = 1, \ldots, M$ are the M the "positive" classes).

The temporal segmentation of the input video is achieved in three steps: discrimination, negative rejection and sequential modeling. Figure 3.28 shows a diagram of the considered method, including typical color-coded representations of the intermediate and final segmentation outputs.

In the discrimination step, each frame of the video F_i is assigned the most probable class y_i^* among the considered M positive classes. In order to perform such assignment, a multi-class classifier trained only on the positive samples is employed to estimate the posterior probability distribution:

$$P(y_i|F_i, y_i \neq 0) \tag{3.1}$$

where $y_i \neq 0$ indicates that the negative class is excluded from the posterior probability. The most probable class y_i^* is hence assigned using the Maximum A Posteriori (MAP) criterion: $y_i^* = \arg\max_{y_i} P(y_i|I_F, y_i \neq 0)$. Please note that, at this stage, the negative class is not considered. The discrimination step allows to obtain a noisy assignment of labels to the frames of the input video, as it is depicted in Figure 3.28. The negative rejection step aims at identifying regions of the video in which frames are likely to belong to the negative class. Since in an egocentric video locations are deemed to change smoothly, regions containing negative frames are likely to be characterized by noisy class assignments. This is expected since the multi-class classifier used in the discrimination step had no knowledge of the negative class. Moreover, consecutive frames of an egocentric video are likely to contain uncorrelated visual content, due to fast head movements, which would lead the multi-class classifier to pick a different class for each negative frame. To leverage this consideration, the negative rejection step quantifies the probability of each frame to belong to the negative class by estimating the variation ratio (a measure of entropy) of the nominal distribution of the assigned labels in a neighborhood of size K centered at the frame to be classified. Let $\mathcal{Y}_i^K = \{y_{i-\lfloor\frac{K}{2}\rfloor}, \ldots, y_{i+\lfloor\frac{K}{2}\rfloor}\}$ be the set of positive labels assigned to the frames comprised in a neighborhood of size K centered at frame F_i. The rejection probability for the frame F_i is computed as the variation ratio of the sample $\mathcal{Y}_i^K$:

$$P(y_i = 0|F_i) = 1 - \frac{\sum_{k=i-\lfloor\frac{K}{2}\rfloor}^{i+\lfloor\frac{K}{2}\rfloor} [y_k = mode(\mathcal{Y}_i^K)]}{K} \tag{3.2}$$

where $[\cdot]$ is the Iverson bracket, $y_k \in \mathcal{Y}_i^K$ and $mode(\mathcal{Y}_i^K)$ is the most frequent label of $\mathcal{Y}_i^K$. Since $y_i = 0$ and $y_i \neq 0$ are disjoint events, the posterior probability defined in Equation (3.1) can be easily merged to the probability defined in Equation (3.2) to estimate the posterior probability $P(y_i|F_i)$. Note that this is a posterior probability over the M positive classes, plus the negative one. The MAP criterion can be used to assign each frame F_i the most probable class y_i using the posterior probability $P(y_i|F_i)$ (see Figure 3.28). Please note that, in this case, the assigned labels include the negative class.

The label assignment obtained in the negative rejection step is still a noisy one (see Figure 3.28). The sequential modeling step smooths the segmentation result enforcing temporal coherence among neighboring predictions. This is done employing a Hidden Markov Model (HMM) [151] with $M+1$ states (M "positive" classes plus the "negative" one). The HMM models the conditional probability of the labels $\mathcal{L} = \{y_1, \ldots, y_N\}$ given the video $\mathcal{V}$:

$$P(\mathcal{L}|\mathcal{V}) \propto \prod_{i=2}^{N} P(y_i|y_{i-1}) \prod_{i=1}^{N} P(y_i|F_i) \tag{3.3}$$

where $P(y_i|I_i)$ models the emission probability (i.e., the probability of being in state y_i given the frame F_i). The state transition probabilities $P(y_i|y_{i-1})$ are modeled defining an "almost identity matrix" which encourages the model to rarely allow for state changes:

$$P(y_i|y_{i-1}) = \begin{cases} \varepsilon, & \text{if } y_i \neq y_{i-1} \\ 1 - M\varepsilon, & \text{otherwise} \end{cases} \tag{3.4}$$

The definition above depends on a parameter ε which controls the amount of smoothing in the predictions. The optimal set of labels $\mathcal{L}$ according to the defined HMM can be obtained using the Viterbi algorithm (see Figure 3.28 - 3. Sequential Modeling). The segmentation $\mathcal{S}$ is finally obtained by considering the connected components of the optimal set of labels $\mathcal{L}$.

3.4.2 Experimental Settings

UNICT-VEDI Dataset

In this section, we discuss the experimental settings used for the experiments performed on the UNICT-VEDI Dataset described in Section 3.2. To assess the potential of the two considered devices, we performed experiments separately on data acquired using HoloLens and GoPro by training and testing two separate models.

To setup the method reviewed in Section 3.4.1, it is necessary to train a multiclass classifier to discriminate between the M positive classes (i.e., environments). We implement this component fine-tuning a VGG19 Convolutional Neural Network (CNN) pre-trained on the ImageNet dataset [68] to discriminate between the 9 considered classes (*Cortile, Scalone Monumentale, Corridoi, Coro di Notte, Antirefettorio, Aula Santo Mazzarino, Cucina, Ventre* e *Giardino dei Novizi*). To compose our training set, we first considered all image frames belonging to the training videos collected for each environment (Figure 3.2). We augment the frames of each of the considered environments including also frames from the training videos collected for the points of interest contained in the environment. We finally select exactly 10000 frames for each class, except for the "Cortile" (1) class which contained only 1171 frames (in this case all frames have been considered). As previously discussed, the same video is used for both the environment "Cortile" (1) and point of interest *Ingresso (1.1).* Therefore, it was not possible to gather more frames from videos related to the points of interest. To validate the performances of the classifier, we randomly select 30% of the training samples to obtain a validation set. Please note that the CNN classifier is trained solely on positive data and no negatives are employed at this stage. To select the optimal values for the parameters K (neighborhood size for negative rejection) and ε (HMM smoothing parameter), we carry out a grid search on one of the test videos, which is used as a validation video. Specifically, we consider $K \in \{50, 100, 300\}$ and $\varepsilon \in [e^{-300}, e^{-2}]$ for the grid search. We select "Test 3" as the validation video for the algorithms trained on data acquired with HoloLens and "Test 4" for the experiments related to data acquired using the GoPro camera. These two videos are selected since they contain all the classes and, overall, similar content (see Table 3.3). Since they have been acquired simultaneously by the same operator to provide similar material for validation. The grid search led to the selection of the following parameter values: $K = 50; \varepsilon = e^{-152}$ in the case of the experiments performed on HoloLens data and $K = 300; \varepsilon = e^{-171}$ for the experiments performed on the GoPro data. We used the Caffe framework [152] to train the CNN models.

All experiments have been evaluated according to two complementary measures: FF_1 and ASF_1 [6]. The FF_1 is a frame-based measure obtained computing the frame-wise F_1 score for each class. This measure essentially assesses how many frames have been correctly classified without taking into account the temporal structure of the predictions. A high FF_1 score indicates that the method is able to estimate the number of frames belonging to a given class in a video. This is useful to assess, for instance, how much time has been spent at a given location, regardless of the temporal structure. To assess how well the algorithm can split the input videos into coherent segments, we also use the ASF_1 score, which measures how accurate the output segmentation is with respect to ground truth. The ASF_1 score is based on the association of each predicted segment to exactly one ground truth segment which is solved using the Hungarian algorithm [153]. FF_1 and ASF_1 scores are computed per class. We also report overall mFF_1 and $mASF_1$ scores obtained by averaging class-related scores.

EGO-CH Dataset

For each cultural site belonging to the EGO-CH Dataset, we trained a VGG-19 CNN to discriminate between locations ("Discrimination" stage). Considering the "Palazzo Bellomo" cultural site, we split the Training Set into two subsets to train and validate the VGG-19 for the "Discrimination" stage (no "negative" frames are used for training). Table 3.20 reports the number of frames belonging to the two subsets for each of the 22 considered environments.

For the "Monastero dei Benedettini" cultural site, we adopted a similar strategy used for the "Palazzo Bellomo". We split the Training Set into two subsets to train and validate the VGG-19 (no "negative" frames are used for training on this cultural site). Table 3.21 reports the number of frames belonging to the two subsets for each of the 4 considered environments.

Table 3.20: Number of frames belonging to the two subsets (Training/Validation) to train the CNN for “Palazzo Bellomo”.

	Training	**Validation**
1 Sala1	2605	1116
2 Sala2	5578	2390
3 Sala3	5800	2486
4 Sala4	8048	3449
5 Sala5	8023	3438
6 Sala6	1841	789
7 Sala7	7429	3184
8 Sala8	4837	2073
9 Sala9	7354	3152
10 Sala10	4081	1749
11 Sala11	5140	2203
12 Sala12	1724	739
13 Sala13	2128	912
14 Cortile degli Stemmi	4097	1756
15 Sala delle Carrozze	2281	978
16 Cortile Parisio	7262	3112
17 Biglietteria	2869	1230
18 Portico	3991	1710
19 Scala Catalana	4479	1920
20 Loggetta	2918	1251
21 Box Sala8	3263	1398
22 Area Sosta	2454	1052
Total	**98202**	**42084**

The “Negative Rejection” step has been considered only for the data of “Monastero dei Benedettini” cultural site, since “Palazzo Bellomo” does not contain negative locations. The “Sequential Modeling” stage allows to obtain a temporal segmentation of the input video where each segment is associated to one of the considered environments. We also evaluated our method on this dataset using FF_1 score and ASF_1 score. For parameter validation purposes (K and ϵ) we used: in “Monastero dei Benedettini” the test video “Test3” and in “Palazzo Bellomo” the Validation set composed by 5 videos. Specifically, $\epsilon = 10^{-273}$ is found by optimizing the validation ASF_1 score with a grid search in the range $[10^{-1} : 10^{-299}]$ on “Palazzo Bellomo”. Since no negative locations are contained in “Palazzo Bellomo”, the “negative rejection” stage is not performed and hence the parameter K is not optimized. Similarly,

Table 3.21: Number of frames belonging to the two subsets (Training/Validation) to train the CNN for “Monastero dei Benedettini”.

	Training	**Validation**
1 Antirefettorio	7000	3000
2 Aula Santo Mazzarino	7000	3000
3 Cucina	7000	3000
4 Ventre	7000	3000
Total	28000	12000

Class	Test1	Test2	Test4	Test5	Test6	Test7	AVG
1 Cortile	0.94	0.00	0.93	0.00	0.95	0.77	0.59
2 Scalone Monumentale	0.99	0.98	0.99	0.95	0.98	0.85	0.96
3 Corridoi	0.93	0.93	0.95	0.85	0.99	0.85	0.92
4 Coro Di Notte	0.94	0.94	0.89	0.87	/	/	0.91
5 Antirefettorio	0.94	/	0.96	0.94	/	0.94	0.95
6 Aula Santo Mazzarino	0.99	/	/	0.98	/	/	0.99
7 Cucina	/	/	0.65	0.75	/	/	0.70
8 Ventre	/	/	0.92	0.99	/	/	0.96
9 Giardino dei Novizi	/	/	0.95	0.71	/	/	0.83
Negatives	0.56	0.50	0.26	0.37	/	/	0.42
mFF_1	0.90	0.67	0.83	0.74	0.97	0.85	0.82

Table 3.22: Frame-based FF_1 scores of the considered method on data acquired using the HoloLens device. The “/” sign indicates that no samples from that class were present in the test video.

we found $\epsilon = 10^{-89}$ and $K = 100$ on “Monastero dei Benedettini”.

3.4.3 Results

In this section, the results obtained for both datasets are reported and discussed in details.

UNICT-VEDI Dataset - Hololens Set

Table 3.22 and Table 3.23 report the mFF_1 and $mASF_1$ scores obtained using data acquired using the HoloLens device.

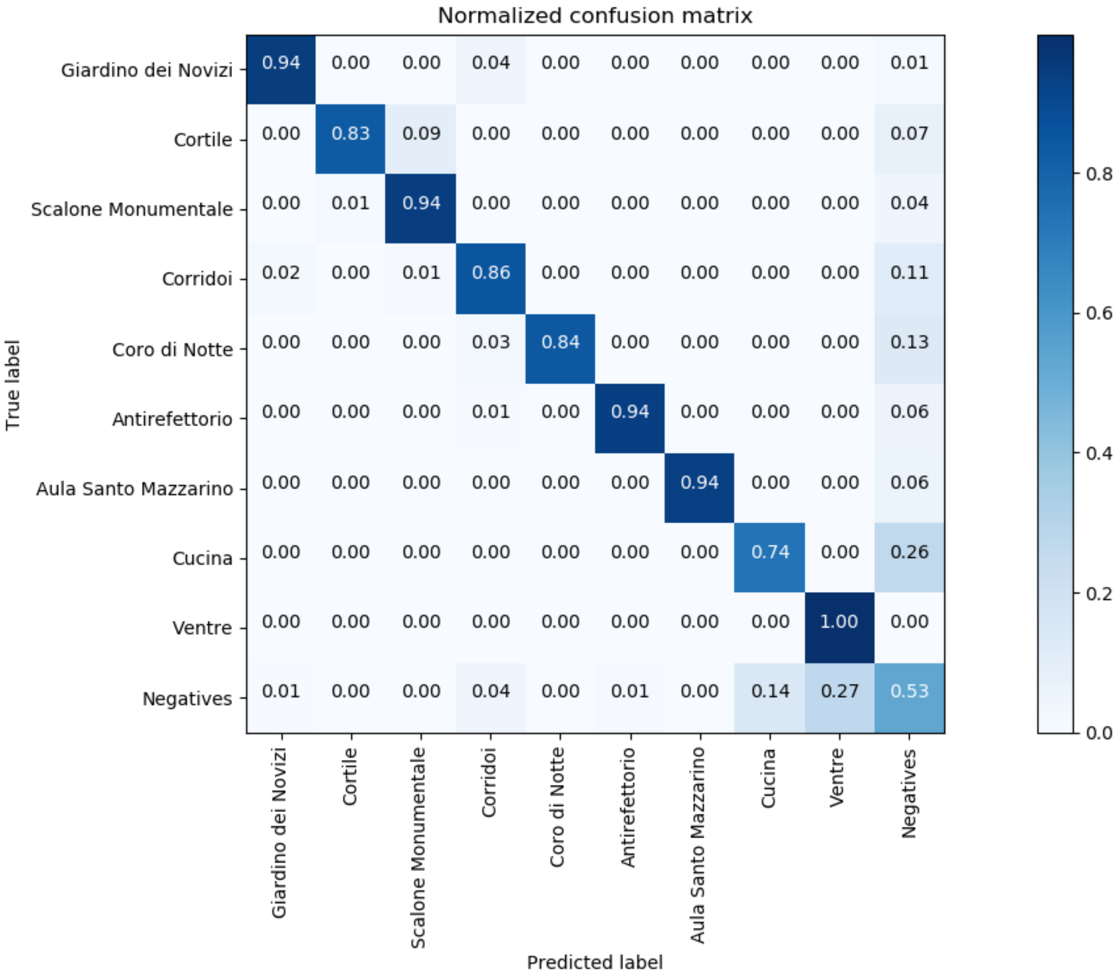
Please note that all algorithms have been trained using only training data acquired with the HoloLens device (no GoPro data has been used). “Test 3” is excluded

Class	Test1	Test2	Test4	Test5	Test6	Test7	AVG
1 Cortile	0.89	0.00	0.86	0.00	0.90	0.62	0.48
2 Scalone Monumentale	0.97	0.95	0.97	0.86	0.97	0.74	0.90
3 Corridoi	0.85	0.87	0.84	0.69	0.99	0.58	0.79
4 Coro Di Notte	0.81	0.90	0.78	0.31	/	/	0.66
5 Antirefettorio	0.88	/	0.93	0.66	/	0.90	0.83
6 Aula Santo Mazzarino	0.99	/	/	0.96	/	/	0.96
7 Cucina	/	/	0.48	0.57	/	/	0.53
8 Ventre	/	/	0.85	0.99	/	/	0.92
9 Giardino dei Novizi	/	/	0.90	0.66	/	/	0.78
Negatives	0.50	0.39	0.12	0.3	/	/	0.27
$mASF_1$	0.84	0.62	0.71	0.60	0.95	0.71	0.71

Table 3.23: Segment-based ASF_1 scores of the considered method on data acquired using the HoloLens device. The "/" sign indicates that no samples from that class were present in the test video.

from the table since it has been used for validation. The tables report the FF_1 and ASF_1 scores for each class and each test video, average per-class FF_1 and ASF_1 scores across videos, overall mFF_1 and $mASF_1$ scores for each test video and the average mFF_1 and $mASF_1$ scores which summarize the performances over the whole test set. As can be noted from both tables, some environments such as "Cortile", "Cucina" and "Giardino dei Novizi" are harder to recognize than others. This is due to the greater variability characterizing such environments. In particular, "Cortile" and "Giardino dei Novizi" are outdoor environments, while all the others are indoor environments. It should be noted that, as discussed before, the two considered measures (mFF_1 and $mASF_1$) capture different abilities of the algorithm. For instance, some environments (e.g., "Corridoi" and "Coro di Notte") report high mFF_1, and lower $mASF_1$. This indicates that the method is able to quantify the overall amount of time spent at the considered location, but temporal structure of the segments is not correctly retrieved. The average $mASF_1$ of 0.71 and mFF_1 of 0.83 obtained over the whole test set indicate that the proposed approach can be already useful to provide localization information to the visitor or for later analysis, e.g., to estimate how much time has been spent by a visitor at a given location, how many times a given environment has been visited, or what are the paths preferred by visitors.

Figure 3.29 reports the confusion matrix of the system on the HoloLens test set.



	Giardino dei Novizi	Cortile	Scalone Monumentale	Corridoi	Coro di Notte	Antirefettorio	Aula Santo Mazzarino	Cucina	Ventre	Negatives
Giardino dei Novizi	0.94	0.00	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.01
Cortile	0.00	0.83	0.09	0.00	0.00	0.00	0.00	0.00	0.00	0.07
Scalone Monumentale	0.00	0.01	0.94	0.00	0.00	0.00	0.00	0.00	0.00	0.04
Corridoi	0.02	0.00	0.01	0.86	0.00	0.00	0.00	0.00	0.00	0.11
Coro di Notte	0.00	0.00	0.00	0.03	0.84	0.00	0.00	0.00	0.00	0.13
Antirefettorio	0.00	0.00	0.00	0.01	0.00	0.94	0.00	0.00	0.00	0.06
Aula Santo Mazzarino	0.00	0.00	0.00	0.00	0.00	0.00	0.94	0.00	0.00	0.06
Cucina	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.74	0.00	0.26
Ventre	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00
Negatives	0.01	0.00	0.00	0.04	0.00	0.01	0.00	0.14	0.27	0.53

Figure 3.29: Confusion matrix of the considered method trained and tested on the HoloLens data.

The confusion matrix does not include frames from the “Test 3” video, which has been used for validation. The matrix confirms how some distinctive environments are well recognized, while others are more challenging. The matrix also suggests that most of the error is due to the challenging rejection of negative samples. Other minor source of errors are the “Giardino dei Novizi - Corridoi”, “Cortile - Scalone Monumentale” and “Coro di Notte - Corridoi” class pairs. We note that the considered pairs are neighboring locations, which suggests that the error is due to small inaccuracies in the temporal segmentation.

Class	Test1	Test2	Test3	Test5	Test6	Test7	AVG
1 Cortile	0.00	0.97	0.95	0.92	/	0.00	0.57
2 Scalone Monumentale	0.92	0.92	0.99	0.99	0.96	0.90	0.95
3 Corridoi	0.90	0.97	0.99	0.99	0.97	0.98	0.97
4 Coro Di Notte	0.89	/	/	/	0.98	0.88	0.92
5 Antirefettorio	/	0.99	0.98	0.96	/	0.87	0.95
6 Aula Santo Mazzarino	/	/	/	/	/	0.90	0.90
7 Cucina	/	/	/	0.89	/	0.83	0.86
8 Ventre	/	/	/	0.99	0.67	0.97	0.88
9 Giardino dei Novizi	/	/	0.99	/	0.95	0.52	0.82
Negatives	0.47	/	/	0.52	0.00	0.21	0.30
mFF_1	0.67	0.96	0.98	0.90	0.76	0.71	0.81

Table 3.24: Frame-based FF_1 scores of the considered method on data acquired using the GoPro device. The "/" sign indicates that no samples from that class were present in the test video.

UNICT-VEDI Dataset - GoPro Set

This section discusses the experiments performed on the GoPro data. To compare the use of different acquisition devices and wearing modalities, we replicate the same pipeline used for the experiments performed on the Hololens data. Hence, we trained and tested the same algorithms on data acquired using the GoPro device.

Table 3.24 and Table 3.25 report the mFF_1 and $mASF_1$ scores for the test videos acquired using the GoPro device. Results related to the "Test 4" validation video are excluded from the tables. The method allows to obtain overall similar performances for the different devices (an average mFF_1 score of 0.81 in the case of GoPro, vs 0.82 in the case of HoloLens and an average $mASF_1$ score of 0.71 vs 0.71). However, $mFF1$ performances on the single test videos are distributed differently (e.g., "Test 1" has a mFF_1 score of 0.90 in the case of HoloLens data and a mFF_1 score of 0.67 in the case of GoPro data).

Figure 3.30 reports the confusion matrix of the method over the GoPro test set, excluding the "Test 4" video (used for validation). Also in this case, errors are distributed differently with respect to the case of HoloLens data. In particular, the confusion between "Cortile" and "Scalone Monumentale" is much larger than in the case of HoloLens data, while other classes such as "Cucina" report better performance on the GoPro data. Moreover, the rejection of negatives is much worse

Class	Test1	Test2	Test3	Test5	Test6	Test7	AVG
1 Cortile	0.00	0.94	0.91	0.85	/	0	0.68
2 Scalone Monumentale	0.85	0.65	0.98	0.97	0.92	0.73	0.85
3 Corridoi	0.86	0.60	0.97	0.99	0.92	0.93	0.88
4 Coro Di Notte	0.76	/	/	/	0.96	0.2	0.58
5 Antirefettorio	/	0.97	0.96	0.92	/	0.66	0.88
6 Aula Santo Mazzarino	/	/	/	/	/	0.81	0.81
7 Cucina	/	/	/	0.79	/	0.68	0.74
8 Ventre	/	/	/	0.99	0.5	0.48	0.66
9 Giardino dei Novizi	/	/	0.97	/	0.91	0.61	0.83
Negatives	0.48	/	/	0.45	0.00	0.24	0.23
$mASF_1$	0.59	0.79	0.96	0.85	0.70	0.53	0.71

Table 3.25: Segment-based ASF_1 scores of the considered method on data acquired using the GoPro device. The "/" sign indicates that no samples from that class were present in the test video.

performing in the case of GoPro data. These differences are due to the different way the GoPro camera captures the visual data. On the one hand, GoPro is characterized by a larger field of view, which allows to gather supplementary information for location recognition. On the other hand, the dynamic field of view of the head-mounted HoloLens device, allows to capture diverse and distinctive elements of the environment and allows for better rejection of negative frames increasing the amount of discrimination entropy in unknown environments.

UNICT-VEDI Dataset Summary

Table 3.26 and Table 3.27 summarize and compare the results obtained training the algorithm on the two sets of data. Specifically, the tables report the average FF_1 and ASF_1 scores obtained in the three steps of the algorithm. As can be noted, significantly better discrimination is overall obtained using GoPro data ($0.88 mFF_1$ vs $0.73 mFF_1$). This is probably due to the wider Field Of View of the GoPro camera, which allows to capture more information about the surrounding environment (see Figure 3.5). Rejecting negative frames is a challenging task, which leads to degraded performances both in the case of frame-based measures (Table 3.26) and segment-based ones (Table 3.27). Interestingly, the negative rejection step works best on HoloLens data ($0.66 mFF_1$ vs $0.54 mFF_1$, and $0.24 FF_1$ vs $0.18 FF_1$ for the

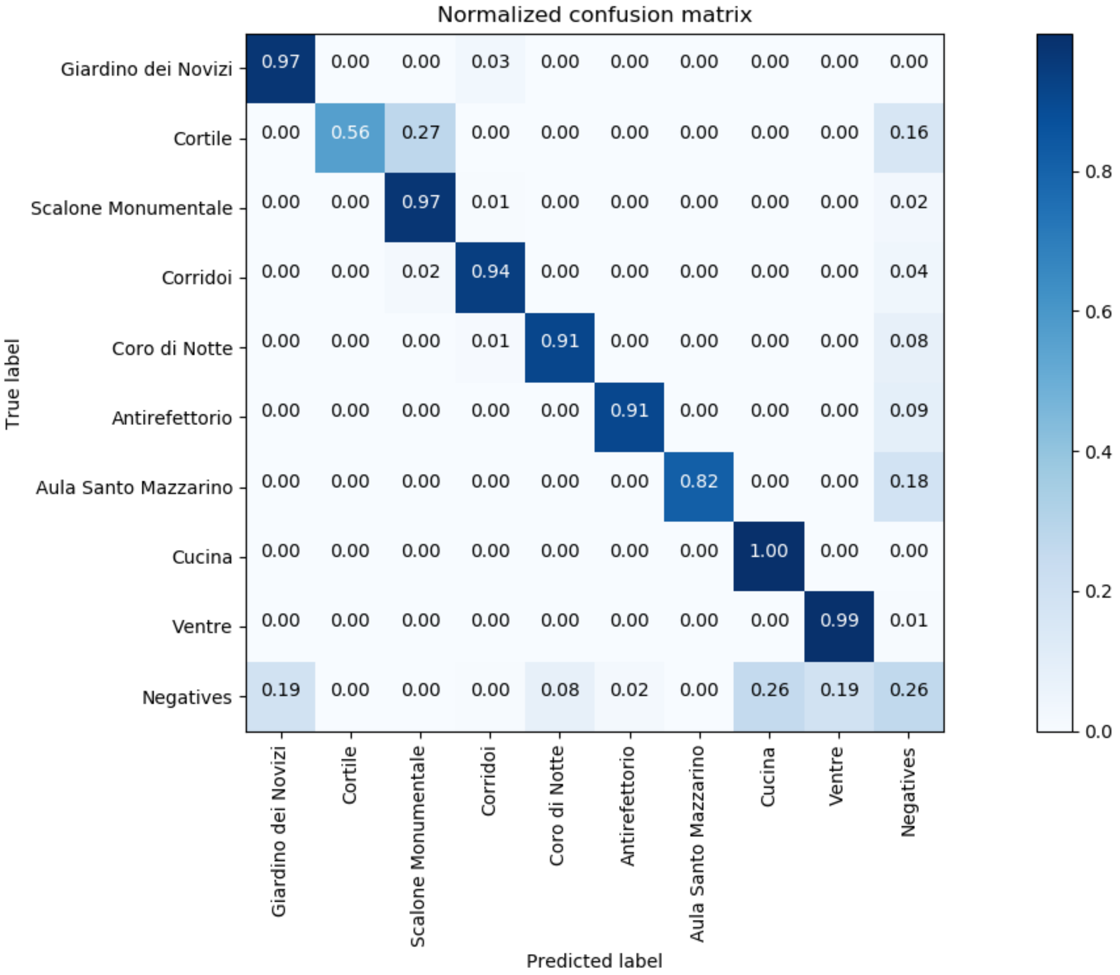


Figure 3.30: Confusion matrix of the results of the considered method on the GoPro test set.

negative class). This result confirms the aforementioned observation that HoloLens data allows to acquire more distinctive details about the scene, thus allowing for more entropy when in the presence of unknown environments. The sequential modeling step, finally balances out the results, allowing HoloLens and GoPro to achieve similar performances.

Figure 3.31 reports a qualitative comparison of the proposed method on “Test3” video (acquired by HoloLens) and “Test4” video (acquired by GoPro) used as validation videos. Please note that the two videos have been acquired simultaneously and so they present similar content. The figure illustrates how the discrimination

	Discrimination		Rejection		Seq. Modeling	
Class	HoloLens	GoPro	HoloLens	GoPro	HoloLens	GoPro
1 Cortile	0.50	0.84	0.45	0.25	0.59	0.57
2 Scalone Monumentale	0.81	0.93	0.84	0.91	0.96	0.95
3 Corridoi	0.77	0.92	0.69	0.83	0.92	0.97
4 Coro Di Notte	0.71	0.91	0.67	0.64	0.91	0.92
5 Antirefettorio	0.66	0.83	0.73	0.62	0.95	0.95
6 Aula Santo Mazzarino	0.69	0.81	0.65	0.23	0.99	0.90
7 Cucina	0.72	0.90	0.60	0.11	0.70	0.86
8 Ventre	0.97	0.99	0.94	0.86	0.96	0.88
9 Giardino dei Novizi	0.79	0.82	0.79	0.78	0.83	0.82
Negatives	/	/	0.24	0.18	0.42	0.30
mFF_1	0.73	0.88	0.66	0.54	0.82	0.81

Table 3.26: Comparative table of average FF_1 scores for the considered method trained and tested on HoloLens and GoPro data. The table reports scores for the overall method (seq. modeling column), as well as for the two intermediate steps of Discrimination and Rejection.

	Discrimination		Rejection		Seq. Modeling	
Class	HoloLens	GoPro	HoloLens	GoPro	HoloLens	GoPro
1 Cortile	0.01	0.15	0.02	0.03	0.48	0.68
2 Scalone Monumentale	0.01	0.05	0.01	0.03	0.90	0.85
3 Corridoi	0.00	0.02	0.00	0.01	0.79	0.88
4 Coro Di Notte	0.00	0.00	0.01	0.01	0.66	0.58
5 Antirefettorio	0.00	0.02	0.01	0.01	0.83	0.88
6 Aula Santo Mazzarino	0.00	0.00	0.00	0.00	0.96	0.81
7 Cucina	0.00	0.01	0.00	0.00	0.52	0.74
8 Ventre	0.00	0.32	0.00	0.03	0.92	0.66
9 Giardino dei Novizi	0.01	0.15	0.01	0.04	0.78	0.83
Negatives	/	/	0.01	0.00	0.27	0.23
$mASF_1$	0.00	0.08	0.01	0.02	0.71	0.71

Table 3.27: Comparative table of average AFF_1 scores for the considered method trained and tested on HoloLens and GoPro data. The table reports scores for the overall method (seq. modeling column), as well as for the two intermediate steps of Discrimination and Rejection.

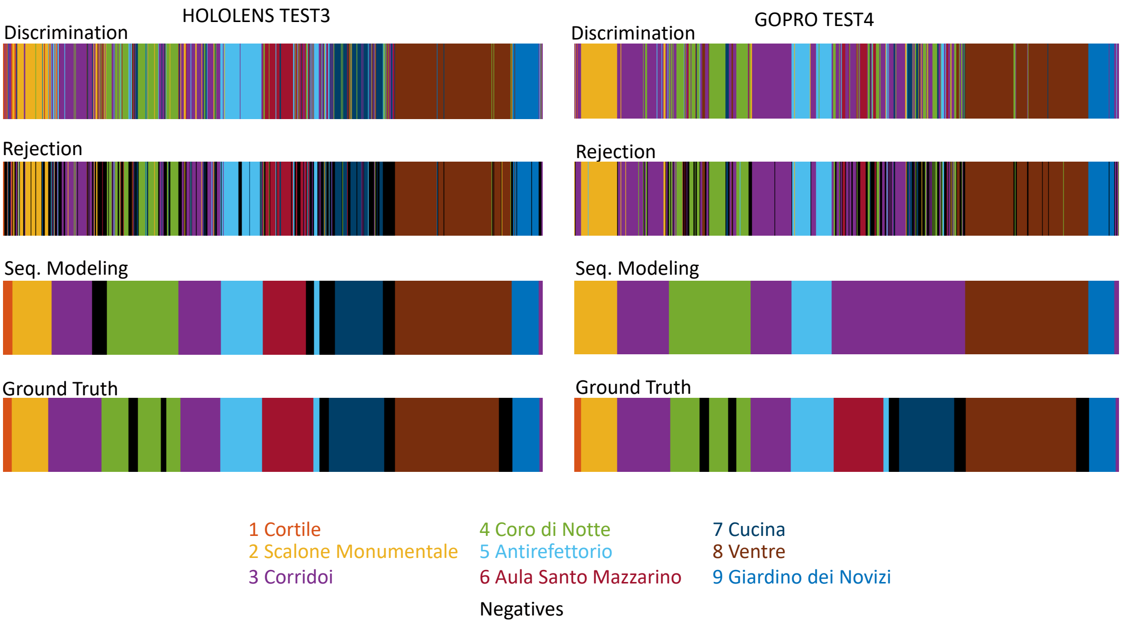


Figure 3.31: Color-coded segmentations for two corresponding test video acquired using HoloLens (left) and GoPro (right).

step allows to obtain more stable results in the case of GoPro data. For this reason, negative rejection tends to be more pronounced in the case of HoloLens data. The final segmentations obtained after the sequential modeling step are in general equivalent.

Even if the final results obtained using HoloLens and GoPro are equivalent in quantitative terms, the data acquired using the HoloLens device is deemed to carry more relevant information about what the user is actually looking at (see Figure 3.6). Such additional information can be leveraged in applications which go beyond localization, such as attention and behavioral modeling. Moreover, head-mounted devices such as HoloLens are better suited then chest-mounted cameras to provide additional services (e.g., augmented reality) to the visitor. This makes in our opinion HoloLens (and head-mounted devices in general) preferable. A series of demo videos to assess the performance of the investigated system are available at our web page `http://iplab.dmi.unict.it/VEDI/video.html`.

Table 3.28: Room-based localization results.For each cultural site, the last row reports the Average (AVG) of the FF_1 and ASF_1 scores.

1) Palazzo Bellomo

Room	FF_1 **score**	ASF_1 **score**
Sala1	0.71	0.48
Sala2	0.92	0.79
Sala3	0.84	0.50
Sala4	0.92	0.59
Sala5	0.94	0.64
Sala6	0.77	0.52
Sala7	0.94	0.61
Sala8	0.89	0.64
Sala9	0.91	0.47
Sala10	0.84	0.69
Sala11	0.84	0.58
Sala12	0.80	0.66
Sala13	0.80	0.66
Cortile degli Stemmi	0.85	0.64
Sala Carrozze	0.91	0.67
Cortile Parisio	0.75	0.50
Biglietteria	0.65	0.44
Portico	0.69	0.51
Scala Catalana	0.76	0.63
Loggetta	0.71	0.51
Box Sala8	0.94	0.79
Area Sosta	0.43	0.47
AVG	**0.81**	**0.59**

2) Monastero dei Benedettini

Class	FF_1 **score**	ASF_1 **score**
Antirefettorio	0.75	0.54
Aula S. Mazzarino	0.33	0.12
Cucina	0.79	0.34
Ventre	0.97	0.60
Negative	0.54	0.33
AVG	**0.68**	**0.40**

EGO-CH Dataset

Table 3.55 reports the results obtained by the baseline in the two cultural sites. On "Palazzo Bellomo", the baseline achieves good FF_1 scores for most rooms, obtaining an average value of 0.81. Much lower results are observed when the ASF_1 score is considered. In this case, an average value of 0.59 is reached. Lower results equal to 0.68 and 0.40 are obtained in the "Monastero dei Bendettini". This is partly due to the presence of negatives, which are not included in "Palazzo Bellomo" and to the more challenging nature of the test set of "Monastero dei Benedettini", which contains 60 videos collected by real visitors within 3 months with different lighting condition and blur as shown in Figure 3.32.

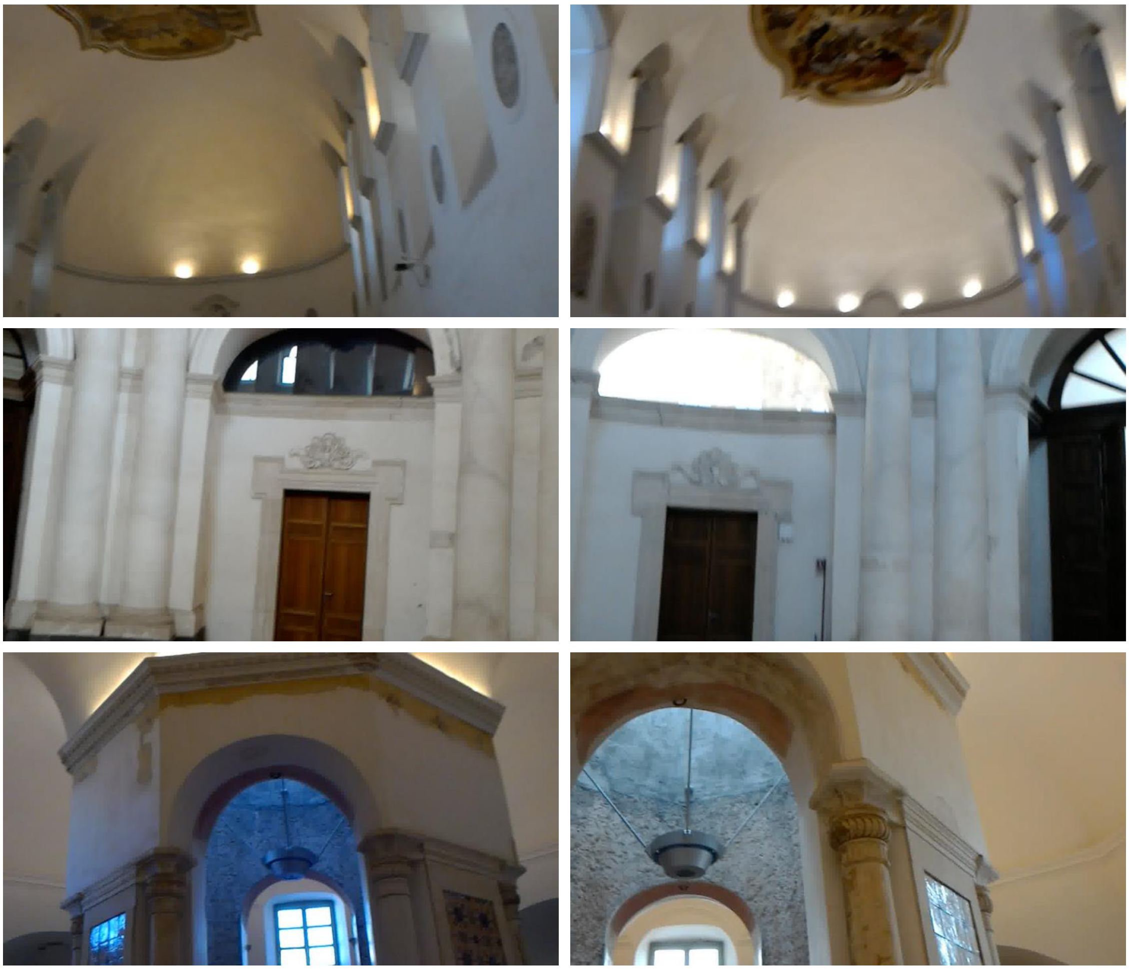


Figure 3.32: Some sample frames from different visits acquired within 3 months. Each row represents similar positions in the same environment with different lighting conditions.

Table 3.29: Detailed results of the 9 test videos of “Palazzo Bellomo” using the FF_1 score. The “/” sign indicates that no samples from that class was present in the test video.

	FF_1 **score**									
Class	Test1	Test2	Test4	Test5	Test6	Test7	Test8	Test9	Test10	AVG
1_Sala1	0,16	0,00	0,81	0,96	0,86	0,96	0,90	0,85	0,92	0,71
2_Sala2	0,78	0,67	0,96	0,96	0,99	0,99	0,97	0,97	0,96	0,92
3_Sala3	0,68	0,75	0,97	0,87	0,72	0,96	0,83	0,89	0,91	0,84
4_Sala4	0,89	0,96	0,93	0,91	/	0,98	0,86	0,91	0,95	0,92
5_Sala5	0,90	0,94	0,98	0,95	0,89	0,95	0,95	0,95	0,97	0,94
6_Sala6	0,84	0,86	0,80	0,59	0,76	0,57	0,93	0,93	0,68	0,77
7_Sala7	0,99	0,95	0,99	0,88	0,84	0,99	0,92	0,93	0,97	0,94
8_Sala8	0,85	0,95	0,96	0,91	0,67	0,90	0,84	0,95	0,93	0,89
9_Sala9	0,86	0,97	0,95	0,90	0,76	0,93	0,94	0,94	0,90	0,91
10_Sala10	0,87	0,96	0,96	0,97	0,00	0,98	0,95	0,97	0,90	0,84
11_Sala11	0,86	0,96	0,97	0,97	0,00	0,97	0,94	0,96	0,96	0,84
12_Sala12	0,82	0,91	0,86	0,94	0,00	0,88	0,96	0,94	0,88	0,80
13_Sala13	0,74	0,94	0,91	0,85	0,00	0,95	0,97	0,92	0,94	0,80
14_CortiledegliStemmi	0,92	0,80	0,93	0,89	0,73	0,92	0,97	0,94	0,57	0,85
15_SalaCarrozze	0,91	0,89	0,96	0,93	0,90	0,90	0,85	0,96		0,91
16_CortileParisio	0,74	0,50	0,92	0,71	0,48	0,91	0,99	0,74	0,72	0,75
17_Biglietteria	0,64	0,79	0,81	0,49	0,74	0,55	0,66	0,61	0,53	0,65
18_Portico	0,71	0,42	0,77	0,70	0,70	0,73	0,71	0,75	0,72	0,69
19_ScalaCatalana	0,70	0,78	0,80	0,77	0,39	0,86	0,83	0,95	0,76	0,76
20_Loggetta	0,62	0,39	0,75	0,58	0,67	0,77	0,84	0,94	0,81	0,71
21_BoxSala8	0,97	0,97	0,98	/	0,79	0,97	0,99	0,94	0,94	0,94
22_AreaSosta	0,24	0,81	0,56	0,46	0,00	0,87	0,56	0,78	0,77	0,56
mFF1	0,76	0,78	0,89	0,82	0,57	0,89	0,88	0,90	0,84	0,81

Table 3.29 and Table 3.30 are an extended version of the previous table considering the $FF1$ score metric and the $ASF1$ score respectively related to the “Palazzo Bellomo” cultural site. As example, Figure 3.33 illustrates qualitatively the segmentation results of the baseline on “Test7” and Figure 3.34 reports the confusion matrix of the baseline on the test set.

The extended versions of the previous table related to the the 60 test video acquired in the“Monastero dei Benedettini” are reported in Table 3.31 and Table 3.32 considering the FF_1 score metric, and in Table 3.33 and Table 3.34 considering the ASF_1 score.

The overall results highlight that addressing the considered task on the proposed dataset is challenging. In particular, issues such as varying lighting conditions and the presence of negatives need to be addressed in task-specific investigations.

DenseNet Backbone We performed experiments using another backbone in the same pipeline to address the room-based localization task. We used DenseNet [154], a densely convolutional connect network which connects each layer to every other layer in a feed-forward fashion. Table 3.35 and Table 3.36 report the results obtained

Table 3.30: Detailed results of the 9 test videos of "Palazzo Bellomo" using the ASF_1 score. The "/" sign indicates that no samples from that class was present in the test video.

				ASF_1						
Class	Test1	Test2	Test4	Test5	Test6	Test7	Test8	Test9	Test10	AVG
1_Sala1	0,07	0,00	0,25	0,92	0,15	0,92	0,62	0,55	0,84	0,48
2_Sala2	0,32	0,46	0,92	0,92	0,97	0,98	0,65	0,94	0,92	0,79
3_Sala3	0,18	0,34	0,66	0,63	0,19	0,88	0,16	0,66	0,83	0,50
4_Sala4	0,58	0,92	0,41	0,45	/	0,95	0,21	0,28	0,89	0,59
5_Sala5	0,57	0,56	0,84	0,77	0,27	0,72	0,49	0,64	0,94	0,64
6_Sala6	0,47	0,61	0,44	0,35	0,32	0,40	0,87	0,80	0,44	0,52
7_Sala7	0,97	0,91	0,97	0,28	0,07	0,97	0,21	0,16	0,94	0,61
8_Sala8	0,40	0,86	0,84	0,59	0,13	0,68	0,70	0,74	0,85	0,64
9_Sala9	0,19	0,93	0,90	0,45	0,32	0,18	0,64	0,14	0,49	0,47
10_Sala10	0,28	0,92	0,92	0,93	0,00	0,96	0,42	0,94	0,81	0,69
11_Sala11	0,21	0,92	0,94	0,94	0,00	0,94	0,13	0,21	0,92	0,58
12_Sala12	0,37	0,82	0,76	0,89	0,00	0,79	0,91	0,63	0,78	0,66
13_Sala13	0,32	0,88	0,83	0,74	0,00	0,90	0,94	0,40	0,89	0,66
14_CortiledegliStemmi	0,78	0,61	0,83	0,79	0,16	0,84	0,72	0,69	0,39	0,64
15_SalaCarrozze	0,49	0,80	0,92	0,87	0,37	0,82	0,17	0,91	/	0,67
16_CortileParisio	0,31	0,22	0,65	0,41	0,16	0,80	0,98	0,47	0,54	0,50
17_Biglietteria	0,41	0,67	0,69	0,32	0,45	0,29	0,38	0,37	0,36	0,44
18_Portico	0,54	0,28	0,60	0,43	0,51	0,58	0,56	0,49	0,58	0,51
19_ScalaCatalana	0,61	0,48	0,66	0,64	0,41	0,72	0,65	0,91	0,62	0,63
20_Loggetta	0,43	0,31	0,37	0,46	0,23	0,63	0,70	0,87	0,64	0,51
21_BoxSala8	0,65	0,94	0,96	/	0,34	0,94	0,98	0,64	0,88	0,79
22_AreaSosta	0,24	0,69	0,42	0,47	0,00	0,76	0,41	0,64	0,62	0,47
mASF1	0,43	0,64	0,72	0,63	0,24	0,76	0,57	0,59	0,72	0,59

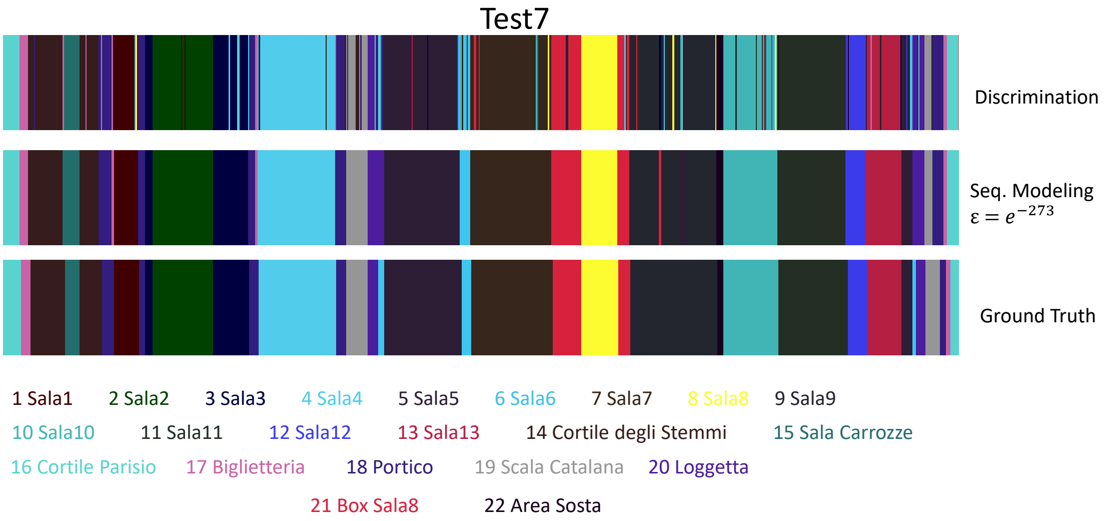
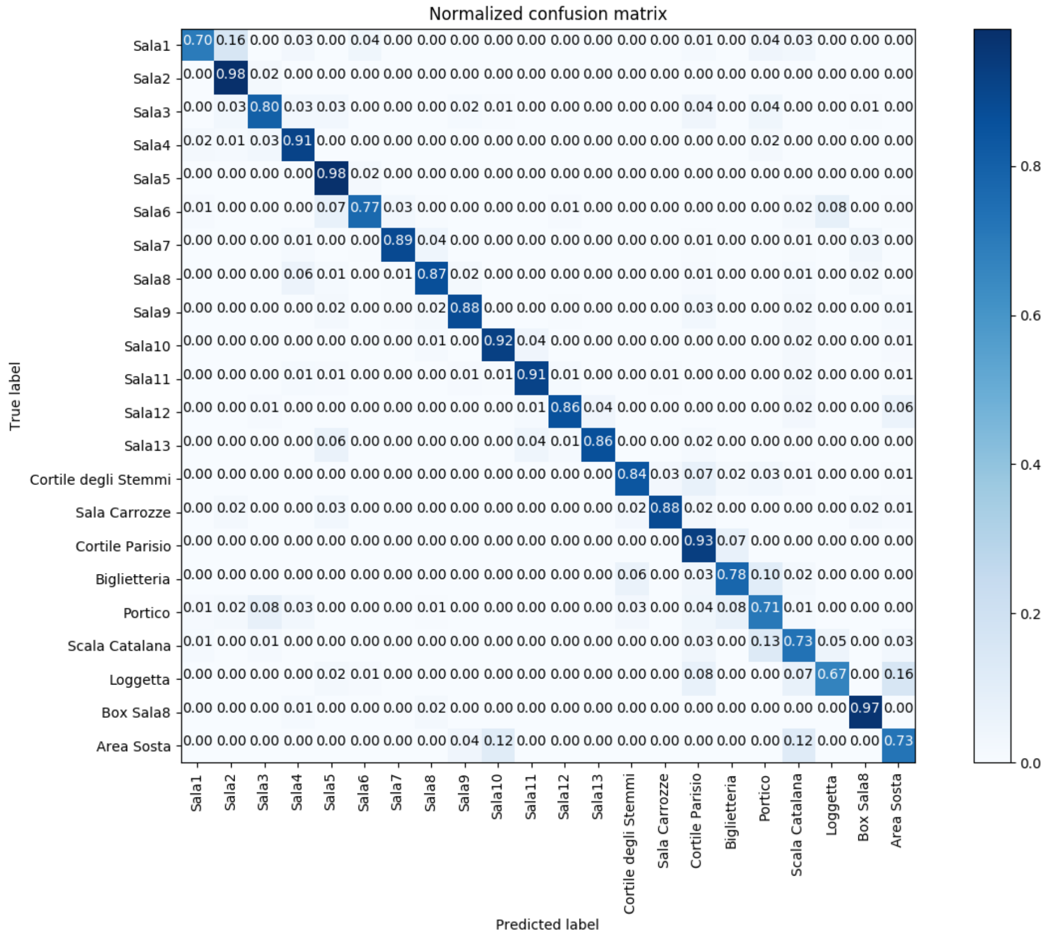


Figure 3.33: Color-coded segmentations for the test video "Test7" of "Palazzo Bellomo".



	Sala1	Sala2	Sala3	Sala4	Sala5	Sala6	Sala7	Sala8	Sala9	Sala10	Sala11	Sala12	Sala13	Cortile degli Stemmi	Sala Carrozze	Cortile Parisio	Biglietteria	Portico	Scala Catalana	Loggetta	Box Sala8	Area Sosta
Sala1	0.70	0.16	0.00	0.03	0.00	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.04	0.03	0.00	0.00	0.00
Sala2	0.00	0.98	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Sala3	0.00	0.03	0.80	0.03	0.03	0.00	0.00	0.00	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.04	0.00	0.00	0.01	0.00
Sala4	0.02	0.01	0.03	0.91	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00
Sala5	0.00	0.00	0.00	0.00	0.98	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Sala6	0.01	0.00	0.00	0.00	0.07	0.77	0.03	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.08	0.00	0.00
Sala7	0.00	0.00	0.00	0.01	0.00	0.00	0.89	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.03	0.00
Sala8	0.00	0.00	0.00	0.06	0.01	0.00	0.01	0.87	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.02	0.00
Sala9	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.02	0.88	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.02	0.00	0.00	0.01
Sala10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.92	0.04	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.01
Sala11	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.01	0.01	0.91	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.02	0.00	0.00	0.01
Sala12	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.86	0.04	0.00	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.06
Sala13	0.00	0.00	0.00	0.00	0.06	0.00	0.00	0.00	0.00	0.00	0.04	0.01	0.86	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00
Cortile degli Stemmi	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.84	0.03	0.07	0.02	0.03	0.01	0.00	0.00	0.01
Sala Carrozze	0.00	0.02	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.02	0.88	0.02	0.00	0.00	0.00	0.00	0.02	0.01
Cortile Parisio	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.93	0.07	0.00	0.00	0.00	0.00	0.00
Biglietteria	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.06	0.00	0.03	0.78	0.10	0.02	0.00	0.00	0.00
Portico	0.01	0.02	0.08	0.03	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.04	0.08	0.71	0.01	0.00	0.00	0.00
Scala Catalana	0.01	0.00	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.03	0.00	0.13	0.73	0.05	0.00	0.03
Loggetta	0.00	0.00	0.00	0.00	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.00	0.00	0.07	0.67	0.00	0.16
Box Sala8	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.97	0.00
Area Sosta	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.04	0.12	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.12	0.00	0.00	0.73

Figure 3.34: Confusion matrix for localization in “Palazzo Bellomo”.

with DenseNet at the end of the Sequential Modeling step. We have evaluated the model using F1 score and Asf1 score. As shown the tables we did not obtain an improvement respect the results obtained with the backbone VGG. In particular for the ”Palazzo Bellomo” we obtained a FF_1 score of 0.71 and a ASF_1 score of 0.58 which are lower respect the scores obtained with VGG ($FF_1 = 0.82$, $ASF_1 = 0.59$). From Table 3.37 to Table 3.40 we report the results obtained in the ”Monastero dei Benedettini” using the backbone DenseNet. As shown, neither considering this cultural site we obtained an improvement of FF_1 and ASF_1 scores.

Table 3.31: Detailed results on the 60 test videos of "Monastero dei Benedettini", considering the evaluation measure FF_1 score. The "/" sign indicates that no samples from that class were present in the test video. The four classes are: 1) Antirefettorio, 2) Aula S. Mazzarino, 3) Cucina, 3) Ventre, whereas Neg. represents the negatives.

ID_Visit	**1**	**2**	**3**	**4**	**Neg.**	**AVG**
4805	0,91	0,33	0,75	0,99	0,43	0,682
1804	0,69	0,43	0,8	0,99	0,43	0,668
4377	0,77	0,47	0,84	0,99	0,5	0,714
1669	0,8	0,51	0,92	0,99	0,67	0,778
1791	0,59	0,22	0,78	0,98	0,35	0,584
3948	0,8	/	0,66	0,99	0,61	0,765
3152	0,71	0,25	0,86	0,98	0,75	0,71
4361	0,89	0,32	0,81	0,099	0,62	0,5478
3976	0,97	0,62	0,92	0,98	0,68	0,834
3527	0,85	0	0,82	0,99	0,63	0,658
4105	0,81	0	0,77	0,99	0,66	0,646
1399	0,65	0	0,76	0,99	0,62	0,604
3836	0,65	0	0,76	0,99	0,62	0,604
4006	0,81	0,82	0,92	0,99	0,75	0,858
4415	0,87	0,49	0,77	0,98	0,73	0,768
3008	0,82	0,2	0,63	0,99	0,23	0,574
4660	0,82	0,2	0,63	0,99	0,23	0,574
2826	0,79	0,57	0,81	1	0,41	0,716
1099	0,77	0,27	0,64	0,98	0,5	0,632
4391	0,8	0,03	0,79	0,98	0,55	0,63
3929	0,94	0	0,84	0,99	0,67	0,688
3362	0,46	0,25	0,76	0,99	0,46	0,584
1379	0,84	0	0,75	0,98	0,33	0,58
2600	0,74	0	0,78	0,99	0,59	0,62
1430	0,57	0,22	0,72	0,96	0,47	0,588
2956	0,53	0,33	0,73	0,96	0,8	0,67
4742	0,14	0,43	0,92	0,99	0,67	0,63
3651	0,94	0,66	0,82	0,99	0,49	0,78
1064	0,77	0,12	0,88	0,99	0,31	0,614
3818	0,93	0,14	0,71	0,99	0,51	0,656

3.5 Points of Interest Recognition

In this Section, we focus on the recognition of points of interest from egocentric images in cultural sites [145],[146]. Recognizing the points of interest observed by visitors in a cultural site is the natural next step after visitor localization [144]. As discussed in Chapter 1, a point of interest can be defined by the site manager as an entity (e.g. object, architectural element, environment etc.) for which it is interesting to estimate the attention of visitors. Figure 3.35 shows some examples of points of interest such as paintings, environments or statues.

We considered the extension of UNICT-VEDI-POIs dataset [145] described in Section 3.2.3 and the EGO-CH dataset[146] described in Section 3.3 to perform experiments to address the task of points of interest recognition. We compare two approaches to recognize points of interest. The first approach is based on the method

Table 3.32: Continued from Table 3.31

ID_Visit	**1**	**2**	**3**	**4**	**Neg.**	**AVG**
2043	0,72	0,47	0,79	0,98	0,66	0,724
3996	0,47	0,4	0,72	0,98	0,76	0,666
3455	0,85	0,33	0,88	0,99	0,61	0,732
4785	0,03	0	0,73	0,96	0,65	0,474
2047	0,95	0,55	0,86	1	0,6	0,792
1912	0,79	0,44	0,67	0,96	0,62	0,696
3232	0,89	0,33	0,82	0,99	0,73	0,752
4442	0,9	0,37	0,82	0,99	0,54	0,724
3646	0,67	0,09	0,8	1	0,34	0,58
4833	0,78	0,59	0,86	0,95	0,66	0,768
3478	0,82	0,51	0,94	1	0,67	0,788
4396	0,81	0,53	0,89	0,99	0,38	0,72
2894	0,67	0	0,87	0,97	0,55	0,612
4414	0,88	0,51	0,9	0,98	0,58	0,77
4639	0,73	0,21	0,83	0,99	0,61	0,674
1004	0,15	0,44	0,6	0,98	0,48	0,53
1917	0,44	0,48	0,51	1	0,42	0,57
1153	0,78	0,54	0,81	0,98	0,57	0,736
2244	0,86	0,3	0,77	0,99	0,32	0,648
2614	0,97	0,59	0,84	0,99	0,39	0,756
1624	0,91	0,71	0,69	0,99	0,48	0,756
3441	0,82	0,45	0,79	0,99	0,46	0,702
4793	0,82	/	0,74	0,99	0,42	0,7425
4083	0,84	/	0,72	0,99	0,73	0,82
4906	0,77	0,2	0,74	0,99	0,42	0,624
1160	0,84	0,66	0,74	1	0,42	0,732
3416	0,77	/	0,82	0,99	0,68	0,815
1051	0,79	0,43	0,78	0,98	0,4	0,676
2580	0,73	0,18	0,89	0,99	0,48	0,654
1109	0,81	0,28	0,89	0,99	0,43	0,68
AVG	**0,75**	**0,33**	**0,79**	**0,99**	**0,54**	**0,68**

described in Section 3.4.1 used for egocentric visitor localization based on a Convolutional Neural Network. It is worth to note that, with this approach, frames are directly processed using a VGG 16 CNN [68] and no object detection is explicitly performed. The second approach is based on an object detector, in particular we considered YOLOv3[61] object detector which is key to obtaining reasonable performance in the recognition of points of interest. Only the latter method has been used on the EGO-CH dataset to perform experiments.

3.5.1 Methods

The first approach implements the method described in Section 3.4.1 which predicts for each input frame, the point of interest observed by the user or the occurrence of the "negative" class to be rejected. This approach has been used only on the UNICT-VEDI dataset. We consider three different variants of this approach

Table 3.33: Detailed results on the 60 test videos of “Monastero dei Benedettini”, considering the evaluation measure ASF_1 score. The “/” sign indicates that no samples from that class were present in the test video. The four classes are: 1) Antirefettorio, 2) Aula S. Mazzarino, 3) Cucina, 3) Ventre, whereas Neg. represents the negatives.

ID_Visit	1	2	3	4	Neg.	AVG
4805	0,71	0,06	0,2	0,31	0,22	0,3
1804	0,55	0,06	0,26	0,15	0,14	0,232
4377	0,55	0,05	0,18	0,98	0,25	0,402
1669	0,4	0,22	0,27	0,93	0,49	0,462
1791	0,46	0,14	0,21	0,25	0,19	0,25
3948	0,67	/	0,26	0,98	0,51	0,605
3152	0,54	0,07	0,66	0,88	0,47	0,524
4361	0,47	0,12	0,17	0,27	0,29	0,264
3976	0,94	0,24	0,45	0,22	0,56	0,482
3527	0,72	0	0,44	0,46	0,55	0,434
4105	0,54	0	0,49	0,39	0,51	0,386
1399	0,15	0	0,44	0,98	0,26	0,366
3836	0,15	0	0,44	0,98	0,26	0,366
4006	0,56	0,69	0,53	0,98	0,5	0,652
4415	0,48	0,15	0,24	0,49	0,48	0,368
3008	0,68	0,1	0,26	0,98	0,1	0,424
4660	0,68	0,09	0,26	0,98	0,09	0,42
2826	0,71	0,14	0,24	0,99	0,41	0,498
1099	0,54	0,15	0,16	0,49	0,31	0,33
4391	0,67	0,05	0,32	0,96	0,53	0,506
3929	0,91	0	0,43	0,98	0,46	0,556
3362	0,36	0,07	0,45	0,57	0,36	0,362
1379	0,49	0	0,21	0,18	0,23	0,222
2600	0,64	0	0,41	0,98	0,38	0,482
1430	0,17	0,09	0,03	0,08	0,1	0,094
2956	0,33	0,06	0,46	0,37	0,48	0,34
4742	0,1	0,09	0,39	0,62	0,17	0,274
3651	0,63	0,19	0,54	0,65	0,28	0,458
1064	0,63	0,05	0,3	0,65	0,13	0,352
3818	0,78	0,11	0,17	0,25	0,35	0,332

57-POI: is the method proposed in [144] and described in Section 3.4.1. The discrimination component of the method is trained to discriminate between the 57 points of interest of the UNICT-VEDI dataset. No “negative” frames are used for training. The rejection of negatives is performed by the rejection component.

57-POI-N: is similar to the 57-POI method, with the addition of a negative class. The discriminator component of the method is trained to discriminate between 57 points of interest plus the “negative” class. In this case, negative frames are explicitly used for training. The rejection component is further used to detect and reject more negatives.

9-Classifiers: nine context-specific instances of the method described in Section 3.4.1 are trained to recognize the points of interest related to the nine different

Table 3.34: Continued from Table 3.33

ID_Visit	1	2	3	4	Neg.	AVG
2043	0,4	0,22	0,2	0,27	0,25	0,268
3996	0,45	0,12	0,39	0,4	0,49	0,37
3455	0,69	0,16	0,39	0,96	0,5	0,54
4785	0,05	0,11	0,53	0,13	0.33	0,205
2047	0,9	0,38	0,22	0,99	0,36	0,57
1912	0,54	0,1	0,16	0,31	0,39	0,3
3232	0,73	0,04	0,49	0,98	0,53	0,554
4442	0,59	0,285	0,22	0,27	0,25	0,323
3646	0,4	0,13	0,2	0,66	0,34	0,346
4833	0,54	0,15	0,53	0,19	0,36	0,354
3478	0,71	0,29	0,54	0,99	0,46	0,598
4396	0,35	0,02	0,05	0,18	0,08	0,136
2894	0,43	0	0,46	0,35	0,18	0,284
4414	0,75	0,11	0,31	0,24	0,3	0,342
4639	0,61	0,12	0,44	0,39	0,43	0,398
1004	0,22	0,11	0,27	0,21	0,38	0,238
1917	0,41	0,11	0,15	0,31	0,26	0,248
1153	0,54	0,11	0,33	0,59	0,26	0,366
2244	0,58	0,16	0,45	0,97	0,26	0,484
2614	0,65	0,05	0,39	0,98	0,3	0,474
1624	0,71	0,29	0,35	0,39	0,23	0,394
3441	0,59	0,13	0,26	0,88	0,25	0,422
4793	0,68	/	0,52	0,98	0,38	0,64
4083	0,65	/	0,31	0,35	0,59	0,475
4906	0,57	0,06	0,33	0,59	0,33	0,376
1160	0,7	0,15	0,34	0,99	0,29	0,494
3416	0,45	/	0,24	0,97	0,19	0,4625
1051	0,55	0,28	0,33	0,4	0,27	0,366
2580	0,37	0,1	0,68	0,21	0,37	0,346
1109	0,66	0,16	0,63	0,98	0,36	0,558
AVG	**0,54**	**0,12**	**0,34**	**0,60**	**0,33**	**0,40**

contexts of the UNICT-VEDI dataset (i.e., one classifier per context). Similarly to 57-POI, no negatives are used for training.

The second approach we consider in our study is based on an object detector. This approach has been considered for both datasets.

Object-based: A YOLOv3[61] object detector is used to perform the detection and recognition of each of the points of interest. We trained YOLOv3 using the standard anchors provided by the authors for the COCO dataset. At test time, YOLOv3 returns the coordinates of a set of bounding boxes with the related class scores for each frame. If no bounding box has been predicted in a given frame, we reject the frame and assign it to the "negative" class. If multiple bounding boxes are found in a specific frame, we choose the bounding box with the highest class-score and assign its class to the frame. We have chosen the YOLOv3 object detector [61] because it is a state-of-the-art real-time object detector.

Table 3.35: Detailed results of the 9 test videos of "Palazzo Bellomo" using the FF_1 score. The backbone used is DenseNet. The "/" sign indicates that no samples from that class was present in the test video.

FF_1 score

Class	Test1	Test2	Test4	Test5	Test6	Test7	Test8	Test9	Test10	AVG
1_Sala1	0,00	0,96	0,96	0,95	0,99	0,94	0,97	0,97	0,91	0,85
2_Sala2	0,82	0,91	0,96	0,95	0,99	0,99	0,99	0,97	0,95	0,95
3_Sala3	0,65	0,84	0,84	0,83	0,94	0,88	0,61	0,85	0,83	0,81
4_Sala4	0,84	0,96	0,98	0,93	/	0,96	0,78	0,97	0,93	0,92
5_Sala5	0,98	0,88	0,96	0,91	0,65	0,91	0,97	0,97	0,99	0,91
6_Sala6	0,84	0,80	0,43	0,00	0,86	0,00	0,18	0,83	0,73	0,52
7_Sala7	0,67	0,93	0,22	0,00	0,38	0,88	0,88	0,40	0,96	0,59
8_Sala8	0,64	0,75	0,60	0,52	0,56	0,56	0,64	0,60	0,77	0,63
9_Sala9	0,90	0,89	0,73	0,88	0,41	0,92	0,98	0,99	0,88	0,84
10_Sala10	0,96	0,98	0,92	0,72	0,00	0,98	0,85	0,97	0,78	0,80
11_Sala11	0,93	0,96	0,96	0,97	0,00	0,97	0,97	0,98	0,95	0,85
12_Sala12	0,82	0,88	0,87	0,90	0,00	0,87	0,85	0,92	0,87	0,78
13_Sala13	0,96	0,95	0,95	0,76	0,00	0,93	0,95	0,94	0,96	0,82
14_CortiledegliStemmi	0,78	0,68	0,74	0,88	0,17	0,90	0,79	0,92	0,00	0,65
15_SalaCarrozze	0,84	0,89	0,95	0,93	0,91	0,84	0,97	0,95	/	0,91
16_CortileParisio	0,33	0,51	0,52	0,45	0,35	0,60	0,81	0,91	0,80	0,59
17_Biglietteria	0,25	0,56	0,00	0,26	0,00	0,00	0,00	0,36	0,00	0,16
18_Portico	0,68	0,67	0,78	0,54	0,69	0,69	0,62	0,78	0,62	0,67
19_ScalaCatalana	0,00	0,00	0,54	0,55	0,78	0,67	0,58	0,85	0,49	0,50
20_Loggetta	0,00	0,50	0,80	0,44	0,72	0,53	0,55	0,83	0,68	0,56
21_BoxSala8	0,96	0,97	0,97	/	0,00	0,97	0,99	0,81	0,94	0,83
22_AreaSosta	0,74	0,85	0,45	0,83	0,00	0,56	0,72	0,85	0,00	0,56
mFF1	0,66	0,79	0,73	0,68	0,45	0,75	0,76	0,85	0,72	0,71

Table 3.36: Detailed results of the 9 test videos of "Palazzo Bellomo" using the ASF_1 score. The backbone used is DenseNet. The "/" sign indicates that no samples from that class was present in the test video.

ASF_1 score

Class	Test1	Test2	Test4	Test5	Test6	Test7	Test8	Test9	Test10	AVG
1_Sala1	0,00	0,93	0,93	0,91	0,97	0,89	0,94	0,94	0,84	0,82
2_Sala2	0,64	0,83	0,91	0,90	0,97	0,97	0,97	0,94	0,90	0,89
3_Sala3	0,36	0,53	0,51	0,53	0,61	0,63	0,29	0,59	0,72	0,53
4_Sala4	0,39	0,92	0,96	0,86	/	0,92	0,31	0,94	0,87	0,77
5_Sala5	0,94	0,64	0,66	0,62	0,34	0,63	0,54	0,95	0,97	0,70
6_Sala6	0,46	0,43	0,18	0,00	0,42	0,00	0,09	0,59	0,51	0,30
7_Sala7	0,50	0,86	0,13	0,00	0,10	0,78	0,28	0,25	0,92	0,42
8_Sala8	0,36	0,44	0,42	0,35	0,35	0,41	0,49	0,53	0,57	0,44
9_Sala9	0,81	0,80	0,51	0,78	0,22	0,84	0,95	0,97	0,78	0,74
10_Sala10	0,92	0,96	0,85	0,19	0,00	0,95	0,27	0,94	0,64	0,64
11_Sala11	0,53	0,92	0,91	0,93	0,00	0,93	0,65	0,95	0,91	0,75
12_Sala12	0,38	0,79	0,76	0,81	0,00	0,77	0,30	0,85	0,76	0,60
13_Sala13	0,93	0,90	0,90	0,62	0,00	0,87	0,91	0,89	0,92	0,77
14_CortiledegliStemmi	0,57	0,53	0,54	0,77	0,08	0,81	0,49	0,67	0,00	0,50
15_SalaCarrozze	0,72	0,80	0,91	0,87	0,83	0,72	0,94	0,90	/	0,84
16_CortileParisio	0,30	0,63	0,43	0,65	0,38	0,47	0,59	0,83	0,66	0,55
17_Biglietteria	0,26	0,43	0,00	0,11	0,00	0,00	0,00	0,39	0,00	0,13
18_Portico	0,51	0,47	0,60	0,45	0,48	0,52	0,42	0,64	0,47	0,51
19_ScalaCatalana	0,00	0,00	0,43	0,51	0,61	0,55	0,42	0,75	0,39	0,41
20_Loggetta	0,00	0,31	0,54	0,28	0,46	0,38	0,48	0,72	0,44	0,40
21_BoxSala8	0,92	0,95	0,95	/	0,00	0,93	0,98	0,67	0,88	0,79
22_AreaSosta	0,59	0,73	0,29	0,71	0,00	0,75	0,57	0,74	0,00	0,49
mFF1	0,50	0,67	0,61	0,56	0,32	0,67	0,54	0,76	0,63	0,58

Table 3.37: Detailed results on the 60 test videos of “Monastero dei Benedettini”, considering the evaluation measure FF_1 score. We used the backbone DenseNet. The “/” sign indicates that no samples from that class were present in the test video. The four classes are: 1) Antirefettorio, 2) Aula S. Mazzarino, 3) Cucina, 3) Ventre, whereas Neg. represents the negatives.

ID_Visit	**1**	**2**	3	4	Neg.	AVG
4805	0,79	0,82	0,74	0,95	0,45	0,75
1804	0,32	0,36	0,79	0,98	0,55	0,6
4377	0,46	0,53	0,8	0,96	0,32	0,614
1669	0,75	0,72	0,9	0,99	0,45	0,762
1791	0,49	0,5	0,84	0,95	0,43	0,642
3948	0	/	0,59	0,98	0,49	0,515
3152	0,39	0,72	0,87	0,97	0,76	0,742
4361	0,55	0,78	0,82	0,97	0,28	0,68
3976	0,87	0,43	0,81	0,93	0,66	0,74
3527	0,86	0,79	0,85	0,99	0,56	0,81
4105	0,62	0	0,77	0,97	0,09	0,49
1399	0,46	0,23	0,79	0,98	0,43	0,578
3836	0,51	0	0,72	0,99	0,55	0,554
4006	0,57	0,78	0,78	0,99	0,37	0,698
4415	0,86	0,61	0,82	0,97	0,35	0,722
3008	0,69	0	0,7	0,99	0,51	0,578
4660	0,62	0,81	0,83	0,98	0,57	0,762
2826	0,31	0,52	0,76	0,97	0,61	0,634
1099	0,62	0,42	0,85	0,98	0,49	0,672
4391	0,74	0,72	0,72	0,98	0,33	0,698
3929	0,26	0	0,8	0,99	0,43	0,496
3362	0,34	0,68	0,68	0,95	0,21	0,572
1379	0,41	0	0,81	0,96	0,45	0,526
2600	0,26	0	0,63	0,96	0,3	0,43
1430	0,94	0	0,69	0,87	0,58	0,616
2956	0,32	0,45	0,33	0,91	0,65	0,532
4742	0,13	0,58	0,59	0,84	0,5	0,528
3651	0,77	0,41	0,88	0,98	0,41	0,69
1064	0,77	0,23	0,83	0,99	0,22	0,608
3818	0,68	0,64	0,73	0,99	0,47	0,702

Table 3.38: Continued from Table 3.37

ID_Visit	**1**	**2**	**3**	**4**	**Neg.**	**AVG**
2043	0,52	0,44	0,62	0,91	0,33	0,564
3996	0,17	0,68	0,57	0,95	0,72	0,618
3455	0,78	0,65	0,8	0,99	0,55	0,754
4785	0,02	0	0,64	0,95	0,33	0,388
2047	0,95	0,84	0,84	0,98	0,38	0,798
1912	0,66	0,51	0,69	0,94	0,46	0,652
3232	0,66	0,53	0,85	0,99	0,49	0,704
4442	0,8	0,77	0,82	0,98	0,23	0,72
3646	0,63	0,29	0,66	0,97	0,4	0,59
4833	0,67	0,82	0,67	0,88	0,24	0,656
3478	0,71	0,75	0,84	0,98	0,37	0,73
4396	0,74	0,69	0,89	0,96	0,26	0,708
2894	0,83	0,8	0,66	0,92	0,48	0,738
4414	0,75	0,7	0,91	0,95	0,22	0,706
4639	0,57	0,52	0,85	0,99	0,39	0,664
1004	0,08	0,72	0,49	0,97	0,26	0,504
1917	0,41	0,71	0,36	0,87	0,29	0,528
1153	0,62	0,7	0,71	0,91	0,32	0,652
2244	0,94	0,56	0,74	0,99	0,43	0,732
2614	0,88	0,56	0,83	0,99	0,4	0,732
1624	0,33	0,8	0,62	0,99	0,33	0,614
3441	0,61	0,25	0,84	0,99	0,41	0,62
4793	0,52	/	0,68	0,99	0,33	0,63
4083	0,93	/	0,73	0,99	0,71	0,84
4906	0,46	0,36	0,59	0,94	0,31	0,532
1160	0,88	0,84	0,72	0,98	0,46	0,776
3416	0,56	/	0,5	0,82	0,2	0,52
1051	0,78	0,76	0,64	0,95	0,57	0,74
2580	0,71	0,45	0,63	0,96	0,44	0,638
1109	0,91	0,28	0,85	0,97	0,36	0,674
mFF1	**0,59**	**0,51**	**0,73**	**0,96**	**0,42**	**0,64**

Table 3.39: Detailed results on the 60 test videos of “Monastero dei Benedettini”, considering the evaluation measure ASF_1 score. We used the backbone DenseNet. The “/” sign indicates that no samples from that class were present in the test video. The four classes are: 1) Antirefettorio, 2) Aula S. Mazzarino, 3) Cucina, 3) Ventre, whereas Neg. represents the negatives.

ID_visit	**1**	**2**	**3**	**4**	**Neg.**	**AVG**
4805	0,55	0,62	0,4	0,12	0,36	0,41
1804	0,32	0,1	0,39	0,05	0,21	0,214
4377	0,31	0,12	0,26	0,08	0,27	0,208
1669	0,68	0,34	0,67	0,9	0,33	0,584
1791	0,41	0,41	0,25	0,06	0,22	0,27
3948	0	/	0,33	0,26	0,49	0,27
3152	0,39	0,43	0,64	0,44	0,58	0,496
4361	0,26	0,35	0,32	0,08	0,2	0,242
3976	0,68	0,26	0,35	0,03	0,5	0,364
3527	0,65	0,37	0,38	0,28	0,44	0,424
4105	0,49	0	0,49	0,16	0,06	0,24
1399	0,55	0,1	0,33	0,23	0,29	0,3
3836	0,47	0	0,35	0,98	0,34	0,428
4006	0,47	0,49	0,49	0,55	0,26	0,452
4415	0,74	0,07	0,35	0,14	0,23	0,306
3008	0,57	0	0,5	0,99	0,41	0,494
4660	0,45	0,52	0,17	0,05	0,27	0,292
2826	0,31	0,29	0,33	0,06	0,25	0,248
1099	0,52	0,24	0,55	0,96	0,17	0,488
4391	0,54	0,13	0,14	0,2	0,24	0,25
3929	0,15	0	0,31	0,22	0,39	0,214
3362	0,2	0,35	0,38	0,05	0,22	0,24
1379	0,22	0	0,43	0,05	0,35	0,21
2600	0,32	0	0,39	0,31	0,26	0,256
1430	0,43	0	0,1	0,02	0,21	0,152
2956	0,41	0,2	0,16	0,29	0,31	0,274
4742	0,15	0,18	0,25	0,11	0,17	0,172
3651	0,32	0,25	0,74	0,48	0,28	0,414
1064	0,63	0,09	0,52	0,97	0,19	0,48
3818	0,43	0,49	0,2	0,06	0,29	0,294

Table 3.40: Continued from Table 3.39

ID_Visit	**1**	**2**	**3**	**4**	**Neg.**	**AVG**
2043	0,4	0,27	0,28	0,13	0,27	0,27
3996	0,23	0,3	0,22	0,05	0,38	0,236
3455	0,63	0,38	0,65	0,99	0,48	0,626
4785	0,06	0	0,25	0,45	0,07	0,166
2047	0,9	0,72	0,72	0,37	0,34	0,61
1912	0,42	0,25	0,2	0,08	0,3	0,25
3232	0,52	0,1	0,34	0,13	0,35	0,288
4442	0,61	0,22	0,33	0,06	0,12	0,268
3646	0,62	0,3	0,26	0,19	0,22	0,318
4833	0,54	0,55	0,26	0,1	0,21	0,332
3478	0,57	0,4	0,47	0,14	0,24	0,364
4396	0,41	0,1	0,2	0,07	0,12	0,18
2894	0,55	0,67	0,36	0,13	0,31	0,404
4414	0,59	0,26	0,83	0,11	0,08	0,374
4639	0,5	0,35	0,67	0,39	0,27	0,436
1004	0,22	0,47	0,17	0,11	0,17	0,228
1917	0,39	0,19	0,03	0,03	0,21	0,17
1153	0,47	0,33	0,21	0,14	0,16	0,262
2244	0,77	0,35	0,49	0,35	0,32	0,456
2614	0,43	0,06	0,39	0,99	0,26	0,426
1624	0,3	0,33	0,24	0,29	0,29	0,29
3441	0,49	0,11	0,44	0,88	0,42	0,468
4793	0,35	/	0,36	0,28	0,37	0,34
4083	0,88	/	0,41	0,25	0,48	0,505
4906	0,36	0,1	0,31	0,06	0,21	0,208
1160	0,79	0,5	0,39	0,14	0,28	0,42
3416	0,32	/	0,25	0,42	0,34	0,3325
1051	0,5	0,31	0,21	0,05	0,4	0,294
2580	0,36	0,29	0,33	0,05	0,28	0,262
1109	0,82	0,17	0,46	0,11	0,28	0,368
mASF1	0,46	0,26	0,37	0,28	0,28	**0,40**

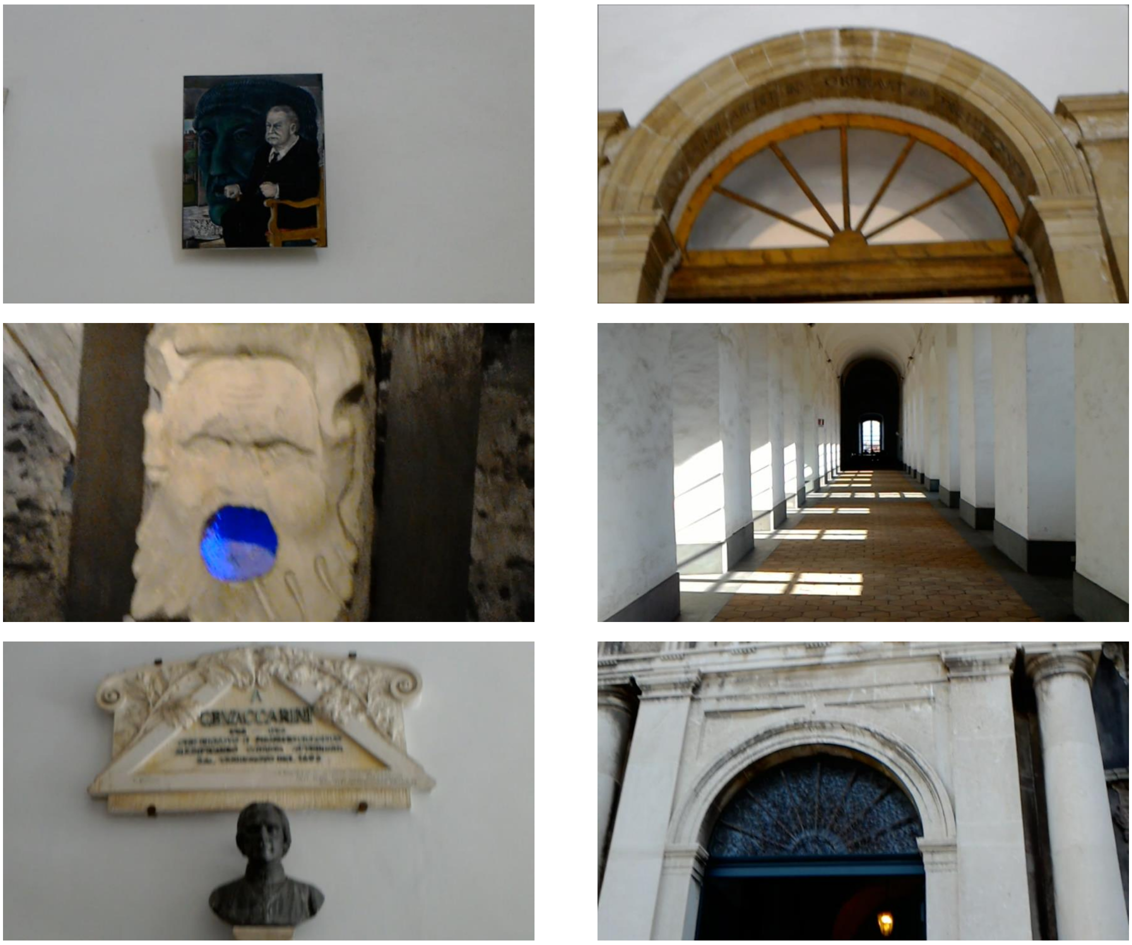


Figure 3.35: Some examples of points of interest: paintings, environments, statues and more. Note that the exhibited variability makes recognition hard.

3.5.2 Experimental Settings

UNICT-VEDI Dataset

In this section, we discuss the experimental settings used for the experiments performed on the UNICT-VEDI-POIs dataset described in Section 3.2.3.

We use mean Average Precision (mAP) with threshold on IoU equal to 0.5 for the evaluations using the method based on the Yolov3 object detector. By default, YOLOv3 only displays objects detected with a confidence score of 0.25 or higher. We performed a validation procedure to optimize this parameter testing the model on the validation video "Test5" using 8 different threshold values (0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4) . We found the best value to be **0.35** for which we obtain a

F_1-score of **0.6751**. To properly compare the approaches described in Section 3.5.1 we use the F_1 score defined as follows:

$$F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{3.5}$$

where precision and recall evaluate the proportion of frames in which points of interest have been correctly detected.

EGO-CH Dataset

Also for the EGO-CH Dataset we use mean Average Precision (mAP) with threshold on IoU equal to 0.5 for the evaluations. In order to use YOLOv3 to detect artworks, a detection threshold is specified to discard detections with low confidence scores. For each cultural site, we tuned this threshold on the validation sets by choosing the value which maximizes mAP in the range $[5^{-4}; 1^{-3}; 5^{-3}; 1^{-2}; 3^{-2}; 5^{-2}; 0.10; 0.15; 0.2; 0.25; 0.3; 0.35; 0.40]$. To train the detector on "Palazzo Bellomo", we set the initial learning rate to 0.001 and the detection threshold to 0.01. On "Monastero dei Benedettini", we set the initial learning rate to 0.01 and the detection threshold to 0.001.

3.5.3 Results

UNICT-VEDI Dataset

Table 3.41 reports the mean average precision (mAP) of YOLOv3 trained on the UNICT-VEDI dataset and tested on the labeled frames of the 7 test videos (2nd column). Table 3.41 also reports the AP scores of some points of interest on which the proposed method obtains the highest performance (3rd - 6th columns) and the lowest performance (7th - 10th columns). The last row shows the average of the (m)AP scores across the test videos. As can be noted from Table 3.41, detecting points of interest is challenging in some cases. In particular, the detector achieves good results for points of interests which represent objects occupying a delimited part of the frame (e.g. see the point of interest 5.5 in Figure 3.8). On the contrary, most of the points of interest where the proposed method has low performance are environments (see for instance the point of interest 3.9 in Figure 3.8). Table 3.42

Table 3.41: Mean Average Precision (mAP) of YOLOv3 on the 7 test videos (2nd column). AP scores are reported for some points of interest (POI) where the proposed method obtains high performances (3rd - 6th columns) and low performances (7th - 10th columns). The last row shows the average of the mAP scores across the test videos. See Figure 3.8 for visual examples of the considered points of interest.

		High performance (AP) on POI x.y				**Low performance (AP) on POI x.y**			
	mAP	**4.2**	**5.5**	**5.10**	**6.2**	**2.1**	**2.2**	**3.9**	**3.11**
Test1	35.04%	49.06%	/	/	100.00%	0.00%	55.81%	12.50%	78.00%
Test2	40.95%	55.41%	/	/	/	56.25%	/	11.96%	/
Test3	47.01%	75.29%	100.00%	81.82%	79.67%	24.62%	12.50%	2.86%	25.74%
Test4	44.60%	66.33%	100.00%	71.43%	/	19.44%	40.08%	12.33%	22.33%
Test5	45.92%	64.29%	100.00%	/	94.74%	80.52%	0.00%	0.00%	10.17%
Test6	24.85%	/	/	/	/	27.47%	6.67%	14.29%	23.64%
Test7	28.84%	/	/	91.67%	/	0.00%	63.21%	12.12%	8.75%
AVG (m)AP	**38.17%**	**62.08%**	**100.00%**	**81.64%**	**91.47%**	**29.76%**	**29.71%**	**9.44%**	**28.11%**

reports the AP values obtained for each class in the 7 test videos. The last row shows the average of the (m)AP scores for each test video.

Table 3.43 compares the three temporal approaches 57-POI, 57-POI-N, 9-Classifiers described in Section 3.5.1 with respect to the approach based on object detection. The second column of Table 3.43 (Discrimination) aims at assessing the abilities of the methods to discriminate among points of interest, in the absence of negatives. In this step, negative frames have been excluded for the evaluation. The rejection step is reported in the third column and includes negative frames for the evaluation. The last column represents the sequential modeling step, where temporal smoothing is applied. This evaluation was performed excluding the "Test5" video which was used for parameter validation purposes.

Among the methods based on the method reported in Section 3.4.1, the one named "9-Classifiers" achieves the best performance in the rejection (F_1-score of 0.64) and sequential modeling steps (F_1-score of 0.66). This highlights the advantages of training separate classifiers for each environment. Only minor improvements are obtained using negatives for training (compare 57-POI with 57-POI-N in Table 3.43). Considering only the positive frames in the Discrimination phase (first column), the object-based method is the best at discriminating the 57 points of interest (F_1 score of 0.78). Analysing the results obtained in the other steps (considering the "negative" frames) the performance obtained by the proposed method is better than the one obtained by the 9-Classifiers approach. Furthermore, the object-based method does not employ any temporal smoothing and the latter is very complex

Table 3.42: Mean Average Precision (mAP) of YOLOv3 on the 7 test videos. AP scores are reported for each point of interest (POI) using a threshold of 0.35.

Class	**Test1**	**Test2**	**Test3**	**Test4**	**Test5**	**Test6**	**Test7**	**AVG**
1.1 Ingresso	73,61%	40,00%	27,27%	53,85%	0,00%	37,50%	35,29%	38,22%
2.1 RampaS.Nicola	0,00%	56,25%	24,62%	19,44%	80,52%	27,47%	0,00%	29,76%
2.2 RampaS.Benedetto	55,81%	/	12,50%	40,08%	0,00%	6,67%	63,21%	29,71%
3.1 SimboloTreBiglie	0,00%	/	0,00%	0,00%	66,67%	0,00%	0,00%	11,11%
3.2 ChiostroLevante	0,00%	/	0,00%	0,00%	35,14%	0,00%	0,00%	5,86%
3.3 Plastico	/	/	/	/	50,00%	/	/	50,00%
3.4 Affresco	0,00%	/	22,73%	6,12%	36,84%	18,46%	0,00%	14,03%
3.5 Fin._ChiostroLev.	0,00%	0,00%	/	0,00%	0,00%	/	/	0,00%
3.6 PortaCorodiNotte	8,89%	16,67%	15,91%	15,79%	7,50%	15,91%	35,90%	16,65%
3.7 TracciaPortone	0,00%	/	/	27,27%	50,00%	57,14%	14,29%	29,74%
3.8 StanzaAbate	/	/	/	/	/	/	/	/
3.9 Corr.DiLevante	12,50%	11,96%	2,86%	12,33%	0,00%	14,29%	12,12%	9,44%
3.10 Corr.CorodiNotte	58,93%	55,32%	61,08%	59,46%	35,77%	72,29%	64,58%	58,20%
3.11 Corr.Orologio	78,00%	/	25,74%	22,33%	10,17%	23,64%	8,75%	28,11%
4.1 Quadro	80,65%	80,00%	47,62%	46,15%	66,67%	/	/	64,22%
4.2 Pav.OriginaleA.	49,06%	55,41%	75,29%	66,33%	64,29%	/	/	62,08%
4.3 BalconeChiesa	40,91%	52,94%	61,82%	/	65,38%	/	/	55,26%
5.1 PortaAulaS.Mazz.	55,41%	/	29,07%	36,36%	20,00%	/	/	35,21%
5.2 PortaIngr.MuseoF.	0,00%	/	33,33%	36,67%	62,50%	/	/	33,13%
5.3 PortaAntirefettorio	0,00%	/	40,91%	9,09%	0,00%	/	/	12,50%
5.4 PortaIng.Ref.Pic.	0,00%	/	66,67%	/	/	/	/	33,34%
5.5 Cupola	/	/	100,00%	100,00%	100,00%	/	/	100,00%
5.6 AperturaPav.	88,89%	/	100,00%	50,00%	/	/	/	79,63%
5.7 S.Agata	100,00%	/	45,83%	50,00%	88,89%	/	/	71,18%
5.8 S.Scolastica	0,00%	/	25,00%	88,89%	97,62%	/	/	52,88%
5.9 ArcoconFirma	/	/	79,69%	100,00%	50,00%	/	49,16%	69,71%
5.10 BustoVaccarini	/	/	81,82%	71,43%	/	/	91,67%	81,64%
6.1 QuadroS.Mazz.	90,00%	/	76,92%	/	92,31%	/	/	86,41%
6.2 Affresco	100,00%	/	79,67%	/	94,74%	/	/	91,47%
6.3 Pav.Originale	56,00%	/	55,56%	/	54,55%	/	/	55,37%
6.4 Pav.Restaurato	13,33%	/	4,17%	/	0,00%	/	/	5,83%
6.5 Bass.Mancanti	13,64%	/	42,01%	/	11,11%	/	/	22,25%
6.6 LavamaniSx	71,43%	/	38,89%	/	0,00%	/	/	36,77%
6.7 LavamaniDx	0,00%	/	38,89%	/	54,44%	/	/	31,11%
6.8 TavoloRelatori	0,00%	/	62,02%	/	0,00%	/	/	20,67%
6.9 Poltrone	39,25%	/	15,54%	/	25,00%	/	/	26,60%
7.1 Edicola	/	/	73,73%	53,85%	65,31%	/	/	64,30%
7.2 PavimentoA	/	/	7,84%	0,00%	15,38%	/	/	7,74%
7.3 PavimentoB	/	/	0,00%	0,00%	37,50%	/	/	12,50%
7.4 Passaviv.Pav.O.	/	/	53,57%	49,12%	43,59%	/	/	48,76%
7.5 AperturaPav.	/	/	28,57%	40,62%	44,74%	/	/	37,98%
7.6 Scala	/	/	70,00%	/	60,00%	/	/	65,00%
7.7 SalaMetereologica	/	/	70,37%	86,21%	26,67%	/	/	61,08%
8.1 Doccione	/	/	23,53%	33,33%	42,59%	/	/	33,15%
8.2 VanoRacc.Cenere	/	/	87,50%	/	100,00%	/	/	93,75%
8.3 SalaRossa	/	/	42,50%	45,24%	61,54%	/	/	49,76%
8.4 ScalaCucina	/	/	61,25%	42,11%	50,76%	/	/	51,37%
8.5 CucinaProvv.	/	/	/	73,33%	82,61%	/	/	77,97%
8.6 Ghiacciaia	/	/	100,00%	/	66,67%	/	/	83,34%
8.7 Latrina	/	/	/	100,00%	50,00%	/	/	75,00%
8.8 OssaeScarti	/	/	68,33%	54,55%	63,16%	/	/	62,01%
8.9 Pozzo	/	/	80,00%	52,08%	85,71%	/	/	72,60%
8.10 Cisterna	/	/	13,89%	53,32%	25,00%	/	/	30,74%
8.11 BustoPietroT.	/	/	67,78%	70,59%	100,00%	/	/	79,46%
9.1 NicchiaePavimento	/	/	45,83%	31,94%	0,00%	/	/	25,92%
9.2 TraccePalestra	/	/	62,50%	70,59%	92,31%	/	/	75,13%
9.3 PergolatoNovizi	/	/	/	60,05%	0,00%	/	/	30,03%
(m)AP	**35,04%**	**40,95%**	**47,01%**	**44,60%**	**45,92%**	**24,85%**	**28,84%**	**38,17%**

Table 3.43: Comparison of the three scene-based approaches and the proposed object-based approach using YOLOv3.

	Discr.	**Reject.**	**Seq. Modeling**
57-POI	0.67	0.55	0.59
57-POI-N	0.53	0.56	0.62
9-Classifiers	0.61	0.64	**0.66**
Object-Based	**0.78**	**0.68**	/

Table 3.44: Object detection results. The reported mean Average Precision (mAP) is averaged over all test videos. Per-class Average Precision (AP) values are reported in the supplementary material.

Cultural Site	**mAP**
1) Palazzo Bellomo	10.59%
2) Monastero dei Benedettini	15.45%

computationally, requiring the optimization of several models in the training phase. It should be noted that, in principle, the results of the object-based method could be further improved introducing some temporal smoothing mechanism, as well as a context-specific approach and rejection mechanism.

EGO-CH Dataset

Table 3.44 reports the results obtained in the two cultural sites of the EGO-CH dataset. The results obtained on "Palazzo Bellomo" are much lower than the ones obtained on "Monastero dei Benedettini" mainly because of the larger set of POIs contained in the former site (191) versus the lower number of POIs contained in the latter (35). In both cases, the results are in general very low, which highlights the challenging nature of the EGO-CH dataset and the considered task. Among the challenges of the dataset, it should be considered that some of the points of interest represent architectural elements such as corridors or pavements, which might be challenging to detect with a simple object detector, as pointed out in [145]. Moreover, differently from other object detection tasks, POIs here need to be recognized at the instance level. For instance, the dataset contains multiple paintings which should be recognized as separate objects.

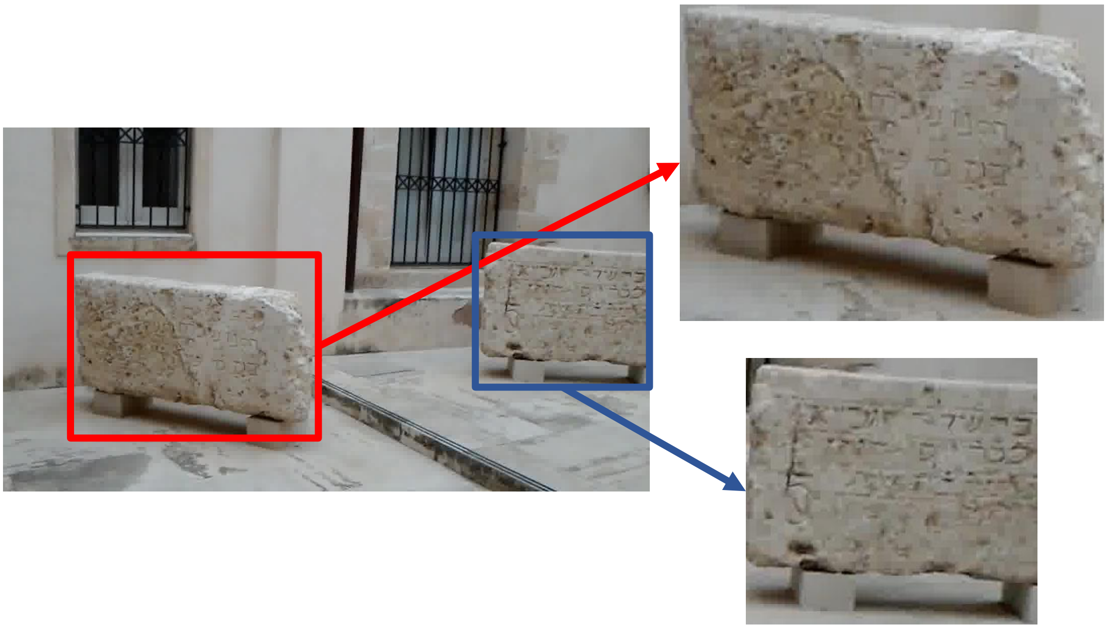


Figure 3.36: Example of image patches extracted using bounding boxes annotations.

3.6 Object Retrieval

Object Retrieval consists in matching an observed object from the egocentric point of view to a reference image contained in the museum catalogue of all artworks. In particular, given a query image containing an object, the task consists in retrieving an image of the same object from a database. This task can be useful to perform automatic recognition of artworks when detection can be bypassed, i.e., when the user places the artwork in the center of the field of view using a wearable or mobile device. Moreover, the task is particularly of interest especially considering that artwork detection is a hard task, as highlighted in Section 3.5. This task has been addressed on the EGO-CH Dataset[146] obtaining a set of query images by extracting image patches from the bounding boxes annotated in the test set. This accounts to *23727* image patches for “Palazzo Bellomo” and *44978* image patches for “Monastero dei Benedettini”. Figure 3.36 shows two examples of image patches extracted from an image labeled with bounding box.

3.6.1 Method

We consider two variants of this task. In the first variant, object retrieval is framed as a one-shot retrieval problem. In this case, the database contains only the reference images associated to each POI, whereas the whole set of image patches is used as the test set, i.e., only a single labeled sample is assumed to be available for each object. In the second variant, we split the set of image patches into a training set (70% - used as DB) and a test set (30%). It should be noted that the first variant of the task is much more challenging both due to the presence of few labeled samples and to the domain shift which affects the two sets of images: reference images for the POIs and image patches cropped from egocentric images.

Given the lack of investigation of approaches for retrieval in the scenario of First-Person vision in the cultural heritage domain, we consider a simple image-retrieval pipeline for both variants of the task. We perform one-shot learning, using a pretrained VGG19 convolutional neural network to extract the features on the high resolution images (Training set) and on the cropped images of the Test set. Then, we train a KNN using only the features belonging to the high resolution images and we tested the algorithm on the Test set. To carry out the standard object retrieval problem, we splitted the test cropped images in Training set and Test set, then we used VGG19 to extract features and KNN to categorize the points of interest. A scheme of the considered baseline is shown in Figure 3.37.

3.6.2 Experimental Settings

We have extracted all features from the FC7 layer of the VGG19 network pre-trained on ImageNet obtaining for each image a 4096-d vector. We used the features extracted from the high resolution images to train a KNN.

Considering the “Palazzo Bellomo” set, for one-shot learning we have used the 191 reference images for training and we tested the trained KNN on 23727 cropped images extracted from the 10 Test videos. Whereas to perform many-shot learning, we used the patches belonging to test videos $1-7$ for training (*15185* patches) and the others to test (*8542* patches). Table 3.45 reports the number of image patches which have been extracted for each test video.

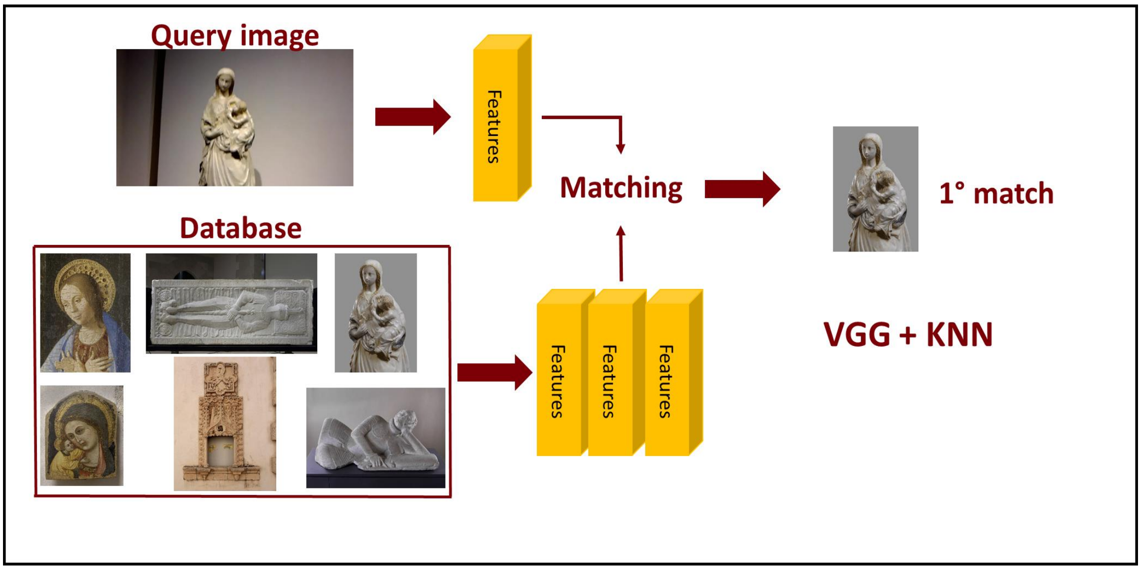


Figure 3.37: Diagram of the baseline for the object retrieval task.

Instead, considering the "Monastero dei Benedettini" set, for one-shot learning we have used as Training set the features extracted from the 35 high resolution images and we categorize with the trained KNN the 44978 cropped images extracted from the 60 real visits. For many-shot learning, we used *30497* image patches images belonging to the visits with IDs from 100 to 147 for training, and *14551* patches belonging to the visits with ID from 148 to 166 for testing. When the second variant of the task is considered, we perform K-NN using $K = [1; 3; 5]$. We evaluated the performance of our baseline using standard metrics for image-retrival: precision, recall and F_1 score. In Table 3.46, we report the number of image patches extracted from the 60 test videos.

3.6.3 Results

Table 3.47 shows the results of the baseline on the image retrieval variants. In both cultural sites, one-shot retrieval does not achieve good results. This is probably due to the fact that one-shot retrieval relies on a limited number of training samples, which are drawn from a different distribution as compared to test samples. This

Table 3.45: Number of patches extracted from each of the 10 test videos of "Palazzo Bellomo".

Test video	**#images**
Test1	2568
Test2	2048
Test3	2672
Test4	2224
Test5	1439
Test6	2086
Test7	2148
Test8	4108
Test9	2914
Test10	1520
Total	**23727**

suggests that dedicated methodologies should be considered to tackle one-shot retrieval and the domain shift problem. Better results are obtained on both sites in the second variant of the task, when the effect of one-shot retrieval and domain shift is reduced. In particular in "Palazzo Bellomo" we obtain (using $K = 1$) the 0.67% considering the F_1 score to categorize 191 points of interest. Instead in "Monastero dei Benedettini" setting $K = 3$ we obtained 0.88% of F_1 score to classify 35 points of interest considering the challenging data of the real 60 visits.

Object Retrieval with DenseNet

We tried to extract features using a different backbone respect to the proposed baseline based on VGG-19. We extracted the features from the FC7 layer of DenseNet[154]. In this way, we obtained for each image a fixed-size vector of 1024 values. We have followed the same pipeline used with VGG to perform object retrieval. Table 3.48 shows the results obtained using DenseNet as backbone. The results show that using DenseNet we didn't obtained an improvement considering the F_1 score as evaluation measure in both variants (One-shot and Many-shots) for both cultural sites.

3.7 Survey Prediction

Each test video of the "Monastero dei Benedettini" is associated to a survey collected from visitors at the end of the visit as described in Section 3.3. We define this task as predicting the content of a survey from the analysis of the related egocentric video. We deem this to be possible as the egocentric video contains information on what the visitor has seen during the visit. In particular, the task consists in predicting for each POI 1) if the POI has been remembered by the visitor and 2) how the POI would be rated by the visitor in a $[-7, 7]$ scale. This task investigates automatic algorithms for automatically "filling in" surveys from videos.

3.7.1 Method

Since the proposed task is novel and very challenging, as a proof of concept, we propose a baseline which takes as input the temporal annotations annotations indicating the objects observed by the visitors in the 60 visits. To obtain fixed-length descriptors for each video, we accumulate the number of frames in which a given POI has been observed in a Bag Of Word representation (see Figure 3.38). In such representation, each component of the fixed-length vector indicates the total time in which a specific point of interest has been observed by the visitor. The vector is hence sum-normalized to reduce the influence of videos with different lengths. The whole training set is normalized with z-scoring and classification is performed using K-NN. We consider two baselines. The first one simply performs a binary classification to predict whether a POI has been remembered by the visitor or not. The second one predicts both if the POI has been seen and what score has been assigned to it. This is tackled as a 15-class classification problem, where class -8 indicates that the POI has not been remembered, whereas the other 14 classes represent the scores from -7 to 7 assigned by the visitors to POIs. We would like to note that we treat the problem as a classification task, as the scores assigned by the visitors are discrete integer numbers. Also, the dataset contains a limited set of data-points, which would prevent the algorithm from generalizing beyond the discrete set of labels available at training time.

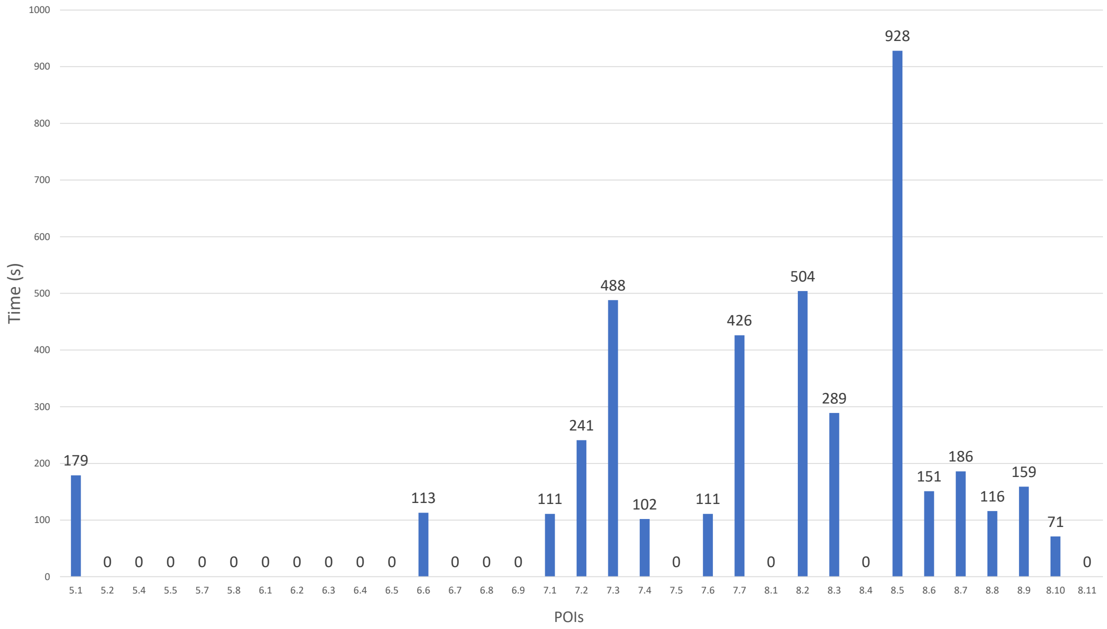


Figure 3.38: An example of the bag-of-word representation used for survey prediction (prior to normalization).

3.7.2 Experimental Settings

We perform our experiments using a leave-one-out strategy. We tested different values for k ranging from 1 to 9 and chose $K = 9$ which resulted to be optimal in our experiments. To perform binary classification we converted the labels equal to -8 in 0 (not remember) and the labels not equal to -8 in 1 (remember). We used a leave one out (LOO) paradigm to train and test the KNN (by varyng K into the range 1-9) over the 60 real visits. In the multiclass classification problem we did not converted the labels. The method adopted to categorize the points of interest into 15 classes has been the same. For both problems we have choosen the optimal $K = 9$. We evaluate results with weighted precision, recall and F_1 score.

3.7.3 Results

Table 3.49 reports the results obtained in the case of binary classification (remembered vs not remembered). The number of instances belonging to each class is reported in the last column. The results suggest that this task is very challenging. Indeed, even if a POI appears in some frames, this does not imply that the visitor remembers it. Table 3.50 shows the results obtained on binary classification using a KNN with different values of k. $K = 9$ gives the best results.

Table 3.51 shows that the multi-class task is even more challenging, with classes containing fewer examples (e.g., $-6, -5, -4, -3$) hard to recognize. As a final remark, it is worth noting that the results suggest that the task can be addressed to some degree. We expect that more complex approaches leveraging the analysis of the semantics of the input videos and the estimation of the attention of the visitor can achieve much better performance. Table 3.52 shows the results of multi-class classification using a KNN classifier with different values of K. We obtained the best results for $K = 9$.

3.8 Semantic Object Segmentation

The aim of a semantic segmentation algorithm is to assign a class label to each pixel of a given input image. Object segmentation retrieves more accurate information compared with standard object recognition. Bounding boxes are not sufficient accurate to fit the shape of objects (e.g, statues, plates, etc.). To this aim, methods required to predict for each pixel of the input image the correct class. We investigated whether the use of both real and synthetic data can help to improve the performance of semantic segmentation algorithms in the context of cultural sites[147]. We performed a preliminary experimental analysis on the extension of the EGO-CH dataset to investigate whether the use of synthetic data can help to improve the performance of semantic segmentation algorithms when applied to real data. This has been tackled by assessing 1) the effect of pre-training the segmentation algorithms on synthetic data, and then fine-tuning them on different amounts of real data; 2) the effect of combining fine-tuning with image-to-image translation to reduce the domain shift between synthetic and real images.

3.8.1 Methods

We compare three approaches to perform object segmentation using real and synthetic data. All approaches consider PSPNet [71] as a baseline semantic segmentation algorithm. PSPNet performs semantic segmentation using a pyramid scene parsing network which extends the features at pixel-level to a designed global pyramid pooling one. The network extracts the feature map from the last convolutional layer, then different sub-region representations are gathered applying a pyramid parsing module followed by upsample and concatenation layers. At the end, the last convolutional layer outputs the final prediction at pixel-level, each pixel belongs into one of the 24 classes of objects.

PSPNet_R This approach consists in training and testing PSPNet on real data. To assess the amount of labeled real data needed to obtain reasonable performance, we train the model with different amounts of training data, namely $5\%, 10\%, 25\%, 50\%$ and 100%.

PSPNet_S+R This approach uses both synthetic and real data during training. The training phase is composed of two stages. In the first stage, PSPNet is trained using only synthetic data. In the second stage, we fine-tune the model obtained at the first stage using real data. The obtained model is then tested on the real images of test set. Similarly to the PSPNet_R approach, we consider different amounts of real data to train the model in the second stage: $0\%, 5\%, 10\%, 25\%, 50\%$ and 100%. The 0% indicates that the model is trained only on synthetic data (first stage training only).

PSPNet_S+R+CycleGAN This approach uses image-to-image translation to reduce the domain shift between real and synthetic images. We have used CycleGAN [85] for this purpose. First, CycleGAN is trained to perform image to image translation between real and synthetic images. Then PSPNet is trained in two stages. In the first stage, it is trained using only synthetic data. In the second stage, real images are transformed to synthetic with CycleGAN and then the network obtained in the first stage is fine-tuned with the traslated images. The model is hence tested on real images transformed to synthetic using CycleGAN. As in the previous cases, we consider different amounts of real training data: $0\%, 5\%, 10\%, 25\%, 50\%$ and 100%.

3.8.2 Experimental Settings

We train all segmentation models with a learning rate of *0.005*, weight decay of *0.0001* and momentum of *0.9*. Each model has been trained for *30* epochs. We selected the best epoch considering the Mean Intersection over Union (*Mean IoU*) on the Validation set. CycleGAN has been trained using the standard parameters suggested in [85]. We evaluated our methods using standard evaluation measures adopted in semantic segmentation benchmarks [30]. The global accuracy (**Accuracy**) counts the fraction of pixels which have been correctly classified. The per-class accuracy (**Class Accuracy**) is the mean of the accuracy values obtained independently for each class. The mean Intersection over Union (**Mean IoU**) is the average of the IoU values between predicted and ground truth segmentation masks, computed independently for each class. The frequency weighted average Jaccard Index (**FWAVACC**) is similar to **Mean IoU**, but per-class IoU values are aggregated using a weighted average based on the number of pixels in each class. While the former two evaluation measures assess the ability to roughly localize objects, the latter two measure how accurate are the predicted semantic masks at the object boundaries.

3.8.3 Results

Table 3.53 and Figure 3.39 report the performances of the three compared approaches on real test data. The results shown in Table 3.53 and Figure 3.39 highlight that using only real data allows to achieve limited performance. Indeed PSPNet_R achieves a maximum Accuracy of 83.51%, a maximum Class Accuracy of 63.15%, a maximum Mean IoU of 47.15%, and a maximum FWAVACC of 72.76%. Using only synthetic data is not sufficient to achieve satisfactory performance on real data (see PSPNet_S+R with 0% real training data in Table 3.53). For instance PSPNet_S+R achieves a Class Accuracy of only 8.45% and a Mean IoU of only 5.50% when trained only on synthetic data. Instead, due to the fact that PSPNet_S+R+CycleGAN was trained using also images belonging to the real domain to learn the synthetic-real translation, it achieves better performance with 0% of real data (Class Accuracy of 53.93% and Mean IoU of 39.43%) respect to PSPNet_S+R.

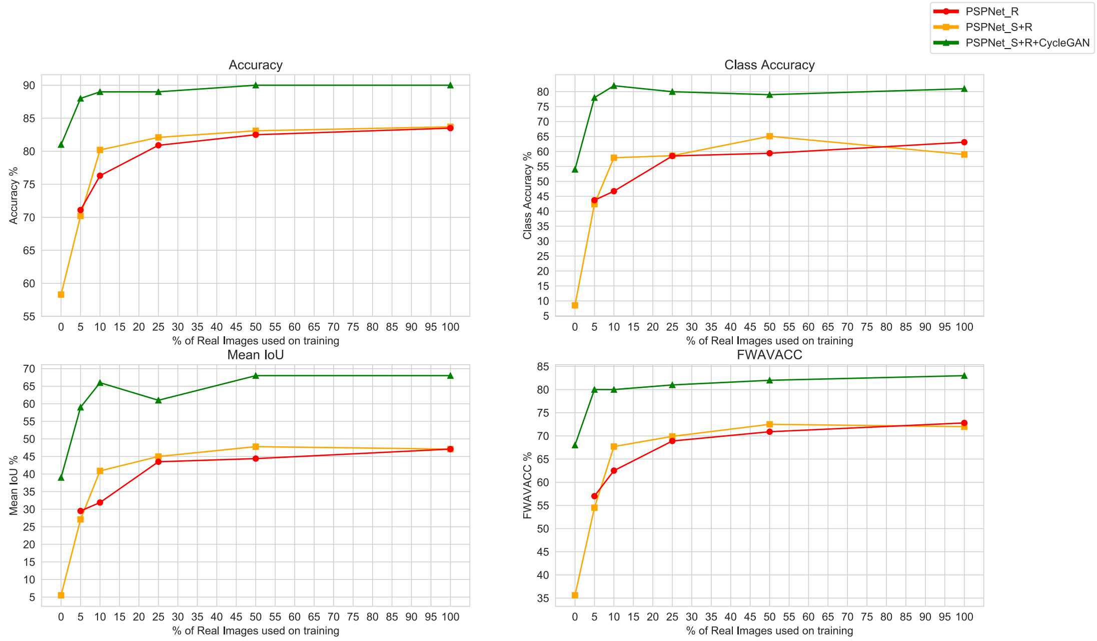


Figure 3.39: Comparison using Real data, Synthetic+Real data, CycleGAN. The range of values in the y axes is different for visualization purpose.

Using 10% of the real data to fine-tune the model (PSPNet_S+R), allows to obtain a Class Accuracy of 57.87% and a Mean IoU of 40.87%, which are comparable to the performances of PSPNet trained on 50% of real data (PSPNet_R), i.e. Class Accuracy 59.40% and Mean IoU 44.37%. This suggests that pre-training with synthetic data helps the model to achieve good performance on real images with less real data for the training procedure. Importantly, when pre-training on synthetic images, we need to label much less data (10% vs 50%) to obtain similar performance. In general, the curves in Figure 3.39 show that PSPNet_S+R needs less real training data to achieve reasonable performance, as compared to PSPNet_R. Adding image-to-image translation to the pipeline (PSPNet_S+R+CycleGAN) allows to outperform all the other approaches even using only 5% of real data for fine-tuning. Indeed, PSPNet_S+R+CycleGAN obtains an Accuracy of 87.82% and a FWAVACC of 79.85% using 5% of real data, which outperforms the best result obtained using 100% of the real data in the PSPNet_R baseline (i.e., 83.51% and 72.76%). Moreover, adding the total amount of real data (100%) PSPNet_S+R+CycleGAN obtains a Class Accuracy of 81.22% and a Mean IoU of 68.20%, which significantly outperform the results achieved by PSPNet_R (63.15% and 47.15%). The curves in Figure 3.39 show that PSPNet_S+R+CycleGAN achieves much better results using the same amounts of real data.

Figure 3.40 reports some qualitative results of the compared approaches. For each example we show the input RGB image, the ground truth segmentation mask and the results obtained by the compared methods when using different amounts of real training data. As can be observed, PSPNet_S+R produces better segmentation masks with less real training data compared to the PSPNet_R, which is trained only using real data (1^{st} example). Moreover, using only 5% of real data (second column) PSPNet_S+R+CycleGAN achieves very accurate segmentation masks as compared to the other two approaches. The second example shows two objects in the scene, but only one of them belongs to the 24 chosen artworks. Both PSPNet_R and PSPNet_S+R wrongly predict a mask for the object in the background, whereas PSPNet_S+R+CycleGAN can segment the correct artwork using only 5% of real data.

Table 3.54 finally reports the results obtained by PSPNet_S+R on the synthetic test data when different amounts of real training data are used to fine-tune the model. As can be noted, fine-tuning greatly impacts performance according to all measures on the synthetic domain. This suggests that the model tends to overfit to the domain of synthetic data during pre-training and similarly, to the domain of real images during fine-tuning. Ideally, semantic segmentation methods should retain good performance on both domains.

3.9 VEDI System

In this section, we present the VEDI (Vision Exploitation for Data Interpretation)[155] system. VEDI is an integrated system which includes a wearable device capable of supporting the visitors of cultural sites, as well as a back-end to analyze the visual information collected by the wearable system and infer behavioral information useful for the site manager. To achieve the aforementioned goals, VEDI

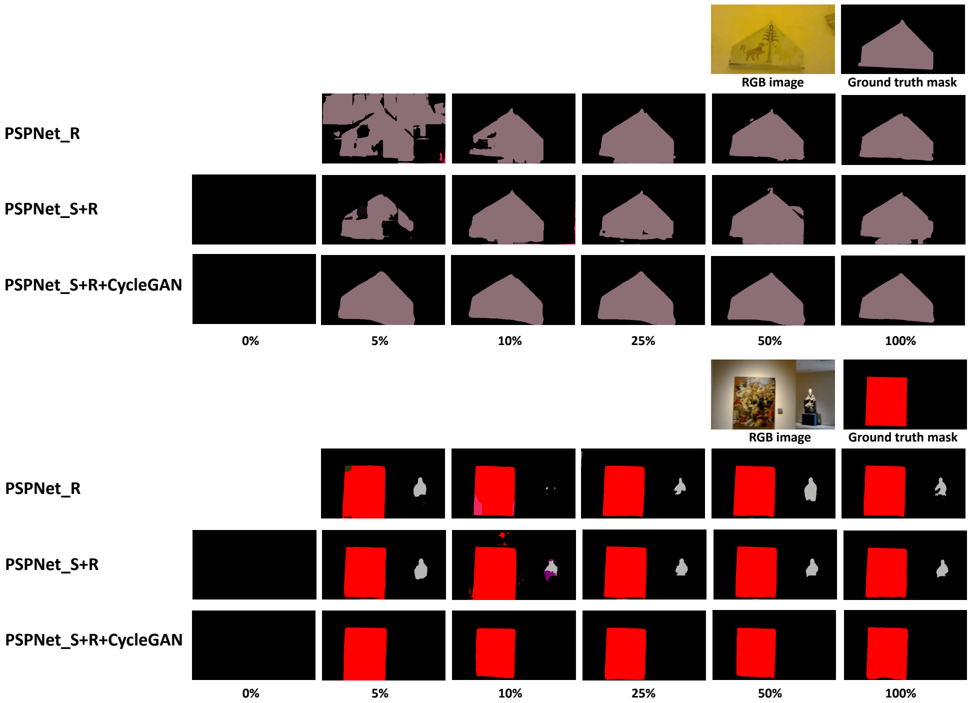


Figure 3.40: Qualitative results of the compared approaches using different amount of real data in the training phase. 0% denotes that the model has been trained using only synthetic images.

uses the algorithms describe in the previous sections. For example, VEDI is able to localize visitors in the cultural site using the algorithm described in Section 3.4 and to recognize the points of interest observed during the visits from the visitors' point of view as detailed in Section 3.5. The inferred information is then used to provide the following services: 1) a "Where am I?" service which informs the visitor on their location in the site during their visit; 2) a service to provide the visitor with additional information on the observed points of interest using Augmented Reality; 3) a service to estimate the visitors' attention during the visits (e.g., what has been seen most, which places have been most visited). The obtained information can be used by the site manager to profile the visitors and gain insights into the quality of the provided services; 4) a recommendation system to suggest visitors what to see next based on their current location and history of observed points of interest; 5) a system to generate a video summary of each visit, which can be given to the visitor as a "digital memory" as reported in Section 3.9.3. Figure 3.41 shows a scheme of the services offered by VEDI to visitors and cultural site mangers.

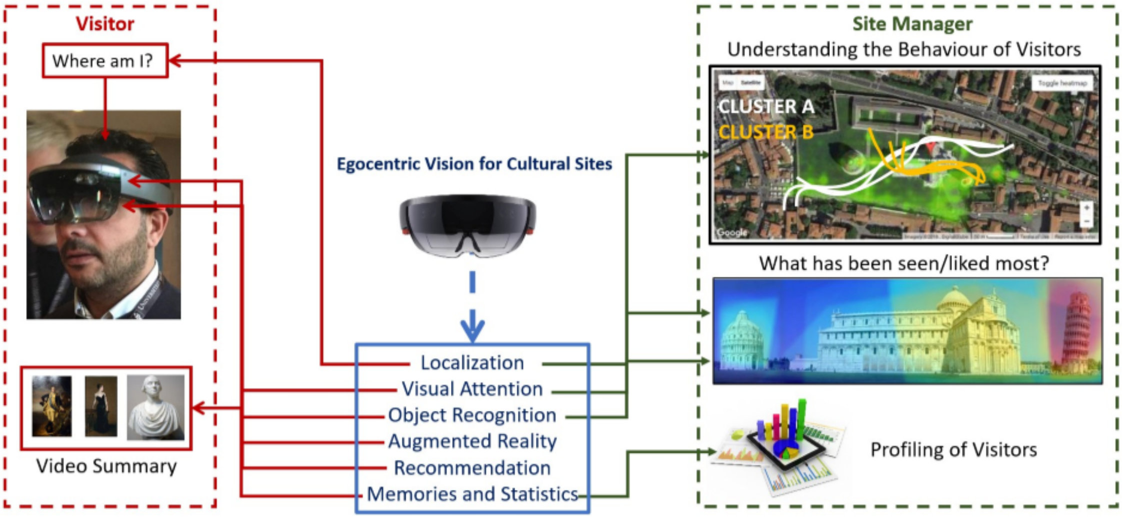


Figure 3.41: The figure shows the architecture of our first person vision system. We reported which services the system provide to the vistors (e.g., Localization, Object Recognition, etc.) and to the cultural manager (e.g., Visual Attention, Statistics, etc.).

To perform experiments with the developed VEDI system, we considered two different cultural sites: the "Monastero dei Benedettini" and the "Orto Botanico" sites both located in Catania. The first one is an indoor environment in which we have considered 9 different environments and 57 different points of interest. This cultural site belongs to the UNICT-VEDI dataset (see Section 3.2). The second one is an outdoor natural site composed by 9 different areas each including plants of different families. We performed experiments related to visitors localization (at room level or area level depending on the site) and to points of interest recognition. The final system has been tested with real visitors showing accurate performances discussed in details in the next sections of this work.

3.9.1 Experimental Cultural Sites and Datasets

The system has been tested considering three different datasets. The UNICT-VEDI-POIs dataset has been described in Section 3.2.3. It was acquired in the "Monastero dei Benedettini"[8], which is an indoor environment. To assess the performances of the localization algorithms in both indoor and outdoor environments we tested

[8] http://monasterodeibenedettini.it

the system on the EgoNature dataset [156]. Moreover, we considered the UNICT-VEDI_Succulente dataset to tackle the problem of points of interest recognition in an outdoor environment. Both EgoNature and UNICT-VEDI_Succulente datasets have been acquired in the "Orto Botanico"[9], which is a outdoor natural site.

UNICT-VEDI-POIs dataset The dataset (described in Section 3.2.3) has been acquired to tackle both room-based localization and points of interest recognition tasks.

EgoNature. The dataset [156] to perform localization in this natural site is composed by 9 contexts and has been acquired using a Pupil 3D Eye Tracker headset and using a smartphone (Honor 9) to collect GPS locations of the user.

UNICT-VEDI_Succulente. This dataset has been collected in the natural site "Orto Botanico" to perform point of interest recognition. It includes 16 points of interest representing plants belonging to following families: 1) Apocynaceae, 2) Bombacaceae, 3) Cactaceae, 4) Crassulaceae, 5) Euphorbiaceae, 6) Lamiaceae, 7) Liliaceae. For each frame, we have annotated the plant depicted in the image. The dataset contains $36,728$ labeled images. Figure 3.42 shows some images of the points of interest present in the dataset.

3.9.2 Architecture and Services

Firstly, the general architecture of VEDI is discussed, then the services implemented by the system are described.

Architecture

Figure 3.43 illustrates the high level architecture of the VEDI system, which is made up of the following components:

- **Mobile devices:** mobile devices such as smart glasses and tablets are provided to the visitors of the cultural site. These devices are used to both acquire images and video from the point of view of the visitors, as well as to provide

[9] http://ortobotanico.unict.it/

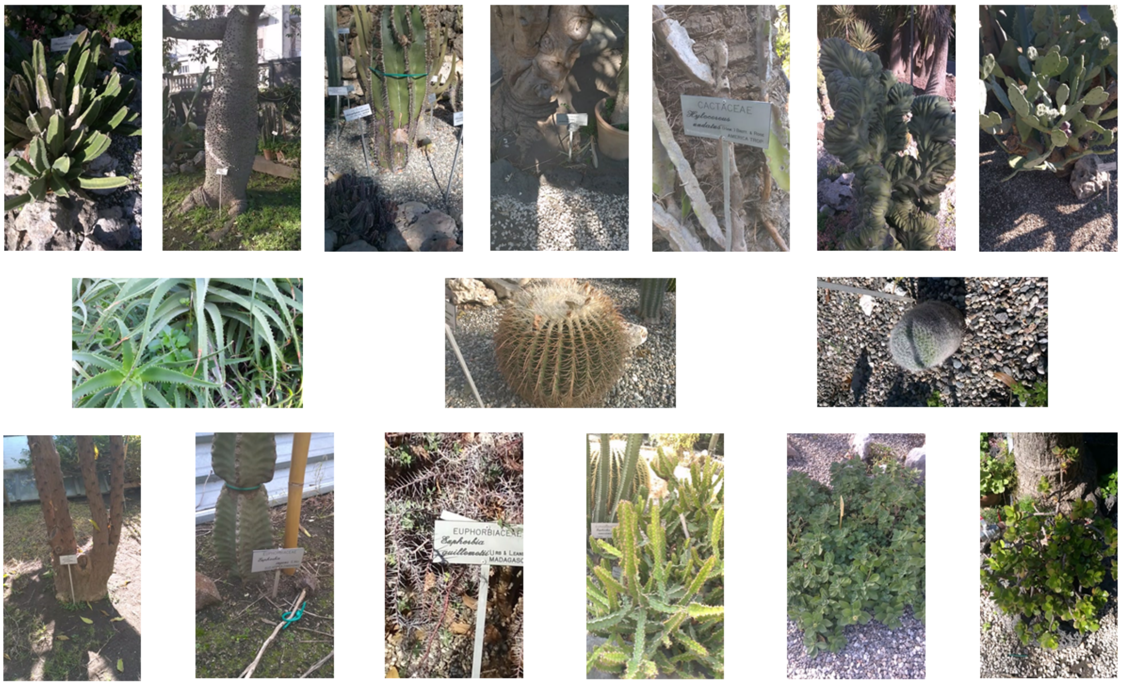


Figure 3.42: Some frames of the plants belonging to the UNICT-VEDI_Succulente dataset.

additional information or recommendations to the visitor through Augmented Reality;

- **Graphic Processing Unit (GPU):** directly connected to the wearable device, it is used to provide additional computational power in order to process egocentric video and address visitor localization and object recognition;
- **Charging and update station:** used at the end of the visit to recharge the wearable devices, transfer the information collected during the visit (e.g., video) to the central system, and update the contents (e.g., 3D models) provided during the visit;
- **Central system:** handles system management, processes and store all data collected by the wearable devices. The central system comprises a *Server*, which includes components to handle the egocentric data collected during the visits and analyze it for behavioral analysis, business intelligence analysis and automatic generation of digital video memories to be provided to the visitors. Moreover, the following actors take part to the central system:

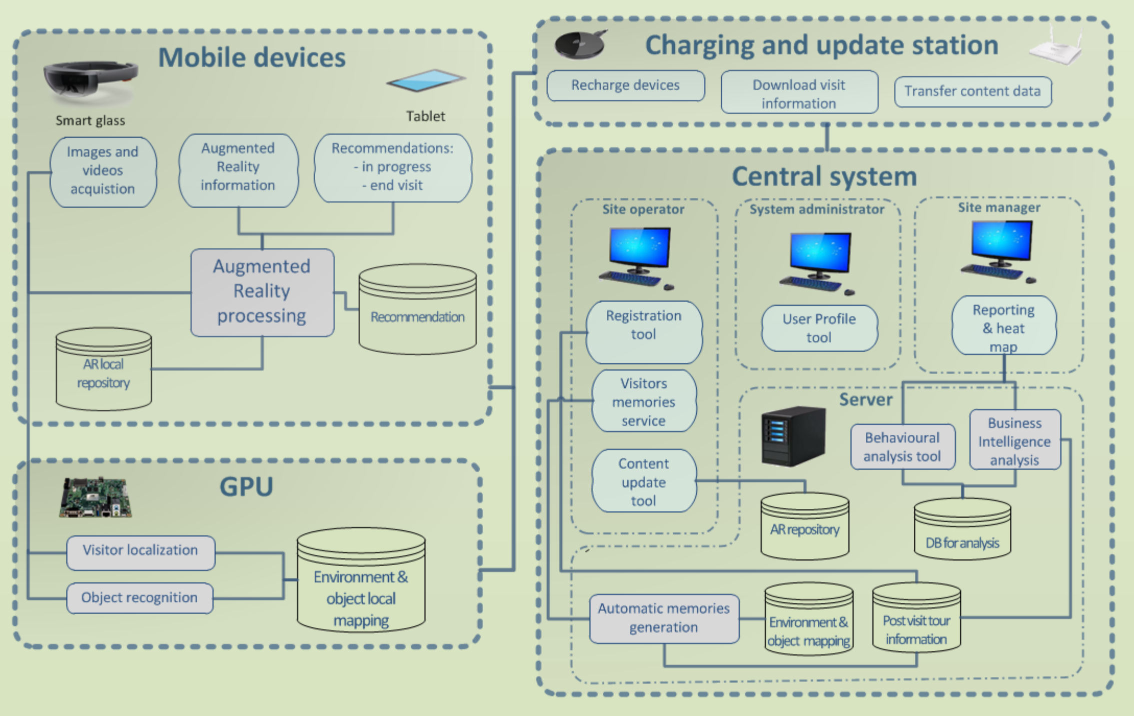

Mobile devices
Smart glass
Tablet
Images and videos acquistion
Augmented Reality information
Recommendations: - in progress - end visit
Augmented Reality processing
Recommendation
AR local repository
GPU
Visitor localization
Object recognition
Environment & object local mapping
Charging and update station
Recharge devices
Download visit information
Transfer content data
Central system
Site operator
System administrator
Site manager
Registration tool
User Profile tool
Reporting & heat map
Visitors memories service
Content update tool
Server
Behavioural analysis tool
Business Intelligence analysis
AR repository
DB for analysis
Automatic memories generation
Environment & object mapping
Post visit tour information

Figure 3.43: The VEDI system is made up of 4 components: 1) Mobile devices, 2) GPU, 3) Charging and update station, 4) Central system.

- *System administrator:* can access all system functions, define user profiles (site operator, site manager) and enable/disable specific functions;
- *Site operator:* can access the following functions: 1) "registration tool", which allows to associate their identity to the assigned mobile device id; 2) "visitors memories service", which automatically generates a video containing the salient moments of each visit to be sent to the user, postcard or other digital gadgets representing objects observed during the visit; 3) "content update tool" which allows to update the contents stored in the AR repository;
- *Site manager:* can use the "Reporting & Head-Map" tool to visualize performance indexes and statistic indicators generated after normalization, aggregation and management of data, as well as all behavioral information periodically extracted by the system using dedicated algorithms.

Services

This Section presents the services implemented by VEDI. Demo videos of the different services are available at the following URL: http://iplab.dmi.unict.it/VEDI_project/#video.

Localization and Points of Interest Recognition: Given the different nature of indoor and outdoor contexts, visitors localization and point of interest recognition are carried out using different algorithms. In indoor contexts (e.g., the *UNICT-VEDI* dataset), the system performs context-based localization of visitors by processing the acquired egocentric video with a multi-stage localization algorithm described in details in Section 3.4. The recognition of the points of interest observed by visitors is carried out as described in Section 3.5. In the *outdoor contexts* (i.e., the *EgoNature* and *UNICT-VEDI_Succulente* datasets), we perform context-based localization by fusing GPS measurements and egocentric images by means of the multi-modal localization algorithm described in [156]. Recognition of points of interest is addressed in *UNICT-VEDI_succulente* as a classification problem, by fine-tuning an AlexNet CNN to discriminate between images belonging to the 16 different points of interest.

Augmented Reality. The AR GUI is triggered when a point of interest is recognized and observed for a significant amount of time. This leaves to the visitor the decision on which "augmented" information they are interested in. To reach this goal, the user interface has been designed according to the following three features:

1. The user interaction panel used to choose the multimedia contents of interest should not remain constantly in front of the visitor;

2. The GUI has been designed relying on the use of transparency to never completely impede the visibility of the external world;

3. The area engaged by the interface is designed to be as small as possible.

See Figure 3.44 for same examples of the AR GUI.

Behavior Analysis and Visual Analytics. To study the behavior of visitors, we compute the following indicators for each cultural site:

- Attraction index: ratio between the number of visitors observing a given point of interest and total number of visitors;
- Retention index: measuring the average time spent in front of an information-communication element (e.g,. a panel, a video, a caption, etc.);
- Usage times: times of use (for the overall visit, for specific sections, for types of users);
- Sweep Rate Index (SRI): the ratio between the total size of the exposure, in square meters, and the average time spent by visitors within the exposure itself;
- Diligent Visitor Index (DVI): the percentage of visitors who stopped in front of more than half of the points of interest of the cultural site.

Data Visualization. The VEDI platform is engineered to provide the managers of cultural sites with utilities and tools to create awareness on the visitors' behavior. Cultural site manager can explore visitors' behavioral data and have insights on the characteristics of each class of visitors (e.g., male-female, young-adult, low-high education, local-alien) through specific data report. This is done relying on the output of the localization and point of interest recognition algorithms discussed in the previous paragraphs. The data visualization tools offer the site manager a way to assess in which areas the visitors spend more time and the most followed routes inside the building (see Figure 3.45).

VEDI assists the managers by providing internationally known key performance indexes such as "Attraction Index", "Sweep Rate Index" and "Diligent Visitors Index" to benchmark the performances of the considered cultural site against similar sites.

Memories. This service allows to automatically generate a video summary of a visit (as described in Section 3.9.3) by taking into account 1) semantic information about locations and observed points of interest obtained using the localization and point of interest recognition algorithms, 2) meta-data (e.g., photos, descriptions) on the site, contexts and points of interest.

3.9.3 Cultural Site Management

The information obtained through the localization's algorithm described in the previous sections, can be used by a site manager to understand where the visitors go during their visits and how much time they spend in each room. In this section, we describe a service of the VEDI system able to collect information useful for the site management. The proposed service [157] is a web tool with a simple Graphical User Interface which is able to summarize each visit producing a "video memory" that can be given as a gift to the visitors so they can share the summary of the visit with others. This module consists of 7 sub-modules which are useful to: 1) create, manage and delete a project related to a cultural site, 2) add rooms of a considered site, 3) define the points of interest for each environment of a site, 4) set the topology of the cultural site, 5) create sample image templates used to create summaries of the visit, 6) generate the videos that summarize the visits, 7) send an email to visitors containing the video summary. Figure 3.46 shows an image of the developed interface. The first four sections of the interface are designed to allow the manager to handle the cultural site (i.e. which environments are there? How many points of interest?), the others are used to automatically generate video summaries of the visits.

Management Interface

In the first section of the interface, called *Projects* (see Figure 3.46, the site manager can create a new project for a cultural site using the button *Create*, delete an existing project through the button *Delete Project* or select the project to manage. For each project, the user can upload a representative *logo* related to the site under consideration. Each site is composed by environments (i.e. a cultural site such as a museum can have a *bookshop*, a *courtyard*, etc.) and the manager can add these using the form called *Environments*. Adding a new environment, the manager is able to insert the name of the considered environment, a description and a map (i.e., an image) which specifies the position of the environments in the current site. Furthermore, the environment can be modified or deleted using the button $Modify-$

Delete Environment. Each environment can have points of interest inside (i.e statues, paintings, etc.) and these data can be included to improve the information about the environment. In the section *Points of Interest*, a point of interest can be added selecting an existing environment. The cultural site manager has to choose a name and the type of the point of interest, insert a description and upload the related picture. As for the environments, is possible to modify and delete an existing point of interest. For each added environment and point of interest, the system assigns a unique identifier (ID). The section *Labeling and Topology* shows a list with all added environments and the corresponding point of interest by using the assigned IDs (Figure 3.47). In the subsection *Topology* it is possible to create the topology of the site as an undirected graph. To create a connection between two environments, the site manager has to enter the IDs of the environments to be connected. Then the topology is generated and shown in Figure 3.48.

Digital Summary

A long egocentric video of a visit is useless for both the visitor and the site manager, due to the huge head motion. Since visitors usually take photos or record short videos to remember or share the most interesting part of a site, our system aims to generate a summary of the video to create a digital gift for the visitors. Assuming to have an egocentric video labeled frame by frame using the method discussed in Section 3.4.1, the system is able to compute a video summary of the environments visited by a tourist. The system takes as input: 1) the descriptions and the maps of environments added in the previous sections *Environments*; 2) the logo of the current project uploaded in the section *Projects*; 3) the image templates automatically generated to describe the environments (see the example in Figure 3.49). The templates are used to create the final video summary. Specifically, for each temporal segment related to an environment, the system associates the related template for n seconds to produce the final video. In the section of the interface called *Video*, the site manager can automatically create the video summary for each visitor and send it via email.

Manager Visualization Tool

Manager Visualization Tool (*MVT*) is an interface useful to help the site manger to analyse the output of the system which automatically localizes (room-based) the visitor during his tour (see Section 3.4.1). With this tool, the site manager, can interact with the segmented videos related to different visits.

The GUI is composed of various sections. The *VideoList* is a list that contains all egocentric videos related to the different visits that the manager can analyse. The section called *Time Spent At Location* is a list that contains all environments present in the selected video. Each environment is represented by a colored block, as shown in Figure 3.50. Each block contains the name of the environment and the time spent by the visitor in that environment. The other main sections of the interface shown in Figure 3.50 are related to the video player and its functionalities. For each frame the interface shows a map that localize the environment of the observed frame and a colored segmented sequence that indicates the predicted labels. Trough a slidebar, the site manager can browse the video. One more section shown in Figure 3.51 is composed of some colored blocks that indicate the frame where a transition phase between the environments start, as well as how much time the visitor spent at that location. Selecting a frame allows to seek the video.

3.9.4 Results

We tested the VEDI system to assess the performances of localization and point of interest recognition systems, which are at the core of VEDI. The method and the results are discussed in details in Section 3.4 and Section 3.5. Table 3.55(a) reports the results of the context-based localization system on UNICT-VEDI dataset while Table 3.55(b) compares the proposed approach for point of interest recognition based on a Yolov3 object detector with respect to the baselines discussed in Section 3.5.

Table 3.55(c) reports the results of different variants of the proposed system for context-based outdoor localization which uses both egocentric images and GPS. Localization results are measured using frame-based accuracy. The time required to process and localize a single image in CPU is reported in milliseconds (ms). All methods use a variant of SqueezeNet to process images and a Decision Tree (DCT) to process GPS. SqueezeNet-n models denote a simplified (and hence faster)

SqueezeNet architecture which considers only the n convolutional layers. All methods obtain good results. Considerably faster inference is obtained using SqueezeNet-6 + DCT. Regarding to the outdoor point of interest recognition system based on AlexNet, it achieves a mean F_1 score of 89.02% on the UNICT-VEDI_Succulente dataset when discriminating among the 16 considered points of interest.

Table 3.46: Number of image patches extracted from the 60 test videos of “Monastero dei Benedettini”.

ID_Visit	#images	ID_Visit	#images
100	770	135	765
101	696	136	414
102	613	137	354
103	1733	138	824
104	768	139	707
105	929	140	494
107	1011	142	770
108	659	143	536
109	234	144	598
110	918	145	897
111	365	146	1307
112	727	147	173
113	288	148	692
114	561	149	954
115	623	150	609
116	968	151	652
117	810	152	699
118	907	153	1244
119	545	154	1691
120	669	155	666
121	774	156	709
122	1156	157	515
123	957	158	846
124	652	159	544
125	491	160	327
126	702	161	985
129	587	162	902
130	820	164	693
132	771	165	1133
134	884	166	690
		Total	**44978**

Table 3.47: Object retrieval results for the two variant of the task.

Points of Interest Retrieval

1) Palazzo Bellomo

Variant	**K**	**Precision**	**Recall**	**F_1 score**
1 - One Shot	1	0.004	0.007	0.001
2 - Many Shots	1	0.69	0.66	**0.67**
	3	0.69	0.62	0.62
	5	0.69	0.62	0.62
	7	0.68	0.62	0.62
	9	0.67	0.61	0.62
	11	0.67	0.61	0.61

2) Monastero dei Benedettini

Variant	**K**	**Precision**	**Recall**	**F_1 score**
1 - One shot	1	0.29	0.07	0.08
2 - Many Shots	1	0.87	0.87	0.87
	3	0.88	0.87	0.87
	5	0.88	0.88	**0.88**
	7	0.88	0.87	0.87
	9	0.87	0.87	0.87
	11	0.87	0.86	0.86

Table 3.48: Results using Densenet.

Points of Interest Retrieval

1) Palazzo Bellomo

Variant	**K**	**Precision**	**Recall**	**F1 score**
1 - One Shot	1	0.02	0.01	0.00
2 - Many Shots	1	**0.62**	**0.59**	**0.6**
	3	0.62	0.56	0.56
	5	0.62	0.56	0.56
	7	0,61	0,56	0,56
	9	0,61	0,55	0,56
	11	0,61	0,55	0,55

2) Monastero dei Benedettini

Variant	**K**	**Precision**	**Recall**	**F1 score**
1 - One shot	1	0.38	0.07	0.09
2 - Many Shots	1	0,83	0,83	0,83
	3	0,84	0,83	0,83
	5	**0,84**	**0,84**	**0,83**
	7	0,84	0,83	0,83
	9	0,84	0,83	0,83
	11	0,83	0,83	0,82

Table 3.49: Survey prediction results - binary classification task.

Class	**Precision**	**Recall**	**F_1 score**	**support**
Not Remembered	0,43	0,2	0,27	561
Remembered	0,74	0,89	0,81	1419
AVG	0,65	0,7	**0,66**	1980

Table 3.50: Results of the binary classifier obtained using a KNN with different values of K.

K	**Precision**	**Recall**	**F1 score**	**Support**
1	0,62	0,61	0,62	1980
3	0,62	0,65	0,63	
5	0,63	0,67	0,64	
7	0,64	0,69	0,65	
9	**0,65**	**0,7**	**0,66**	

Table 3.51: Survey prediction results - multi-class classification. "Weighted AVG" reports the average scores weighted by the number of samples in each class.

Class	**Precision**	**Recall**	**F_1 score**	**Support**
Not Remem.	0,32	0,63	0,43	561
-7	0,52	0,24	0,33	49
-6	0	0	0	8
-5	0	0	0	8
-4	0	0	0	5
-3	0	0	0	5
-2	0,09	0,08	0,08	13
-1	0	0	0	10
0	0,18	0,15	0,17	104
1	0	0	0	36
2	0,02	0,02	0,02	65
3	0,12	0,02	0,04	91
4	0,1	0,04	0,06	181
5	0,13	0,07	0,09	213
6	0,14	0,09	0,11	248
7	0,33	0,29	0,31	383
weighted AVG	0,23	0,27	**0,23**	1980

Table 3.52: Results of the multi-class classifier obtained using a KNN with different values of K.

K	**Precision**	**Recall**	**F1 score**	**Support**
1	0,2	0,2	0,2	1980
3	0,2	0,23	0,19	
5	0,2	0,24	0,21	
7	0,22	0,24	0,22	
9	**0,23**	**0,27**	**0,23**	

Table 3.53: Results of the compared methods on real test data.

	Real Training Data	**Accuracy%**	**Class Accuracy%**	**Mean IoU%**	**FWAVACC%**
PSPNet_S	5%	71.10	43.73	29.47	56.96
	10%	76.28	46.66	31.88	62.49
	25%	80.95	58.54	43.47	68.86
	50%	82.47	59.40	44.37	70.86
	100%	83.51	63.15	47.15	72.76
PSPNet_S+R	0%	58.32	08.45	05.50	35.60
	5%	70.18	42.38	27.06	56.54
	10%	80.23	57.87	40.87	67.71
	25%	82.14	58.55	45.03	69.90
	50%	83.07	65.09	47.80	72.51
	100%	83.70	59.02	47.06	72.00
PSPNet_S+R+CycleGAN	0%	80.52	53.93	39.43	67.77
	5%	87.82	77.90	59.49	79.85
	10%	88.58	81.67	66.19	80.45
	25%	88.62	79.91	60.93	80.57
	50%	90.23	78.72	68.25	82.44
	100%	90.23	81.22	68.20	82.77

Table 3.54: Results of PSPNet_S+R on the synthetic data

	chunk	**Accuracy%**	**Class Accuracy%**	**Mean IoU%**	**FWAVACC%**
PSPNet_S+R	0%	95.31	88.10	81.48	91.20
	5%	77.88	58.39	39.14	69.28
	10%	80.32	60.94	43.44	71.44
	25%	86.55	63.76	50.15	77.48
	50%	83.15	62.93	47.08	74.33
	100%	86.63	59.64	49.35	76.90

	HoloLens	**GoPro**
mFF_1	0.82	0.81
$mASF_1$	0.71	0.71

(a)

Method	F_1 **score**
57-POI	0.59
57-POI-N	0.62
9-Classifiers	0.66
Proposed	0.68

(b)

Method	**Accuracy**	**Time (ms)**
SqueezeNet-6 + DCT	0.86	4.7
SqueezeNet-9 + DCT	0.86	6.09
SqueezeNet-11 + DCT	0.86	6.60
SqueezeNet + DCT	0.91	22.9

(c)

Table 3.55: The results obtained by VEDI system in the fundamental tasks of localization (a and c) and point of interest recognition (b).

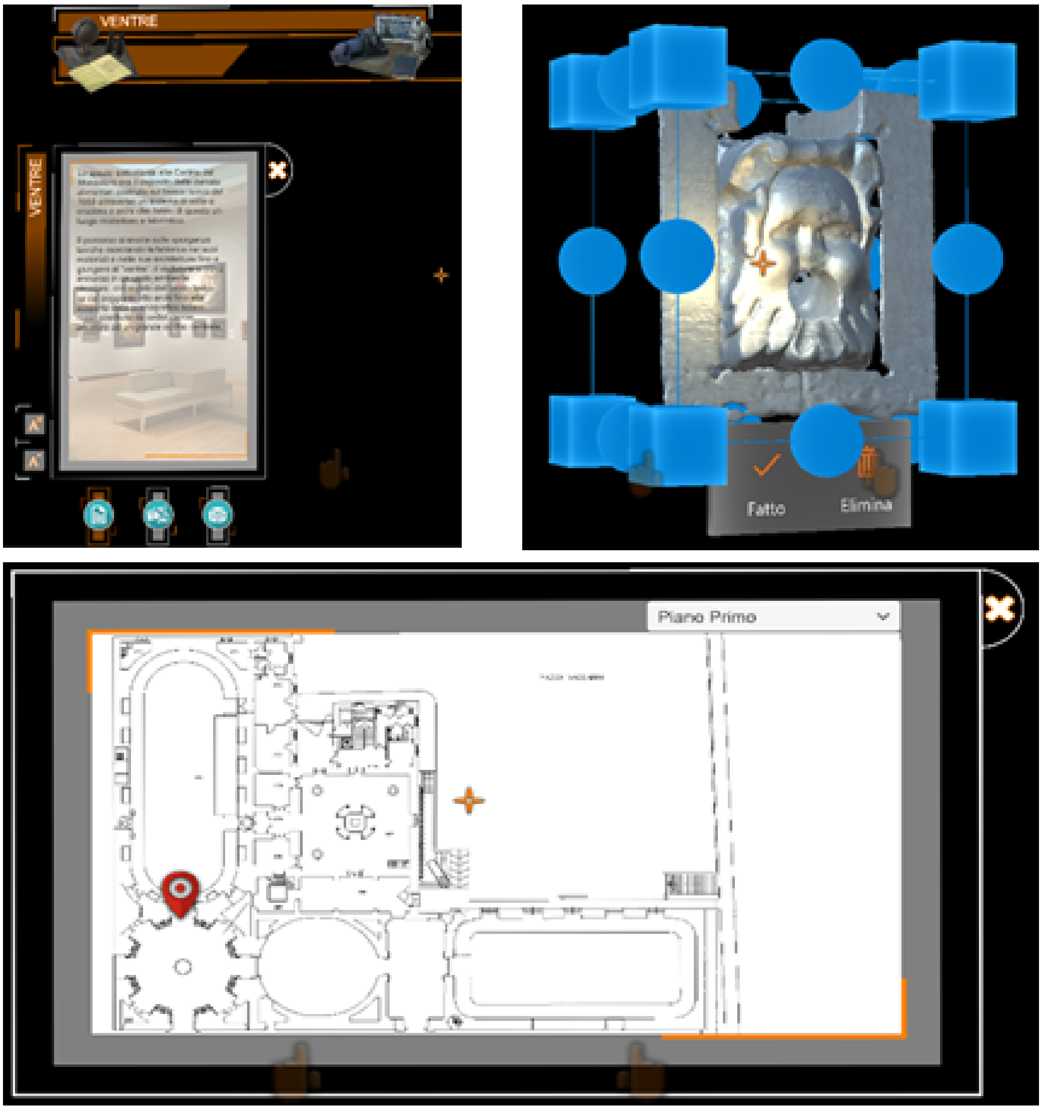


Figure 3.44: AR GUI examples. From left to right: additional information on the observed point of interest, a 3D model shown to the visitor, a map showing the position of the visitor in the cultural site.

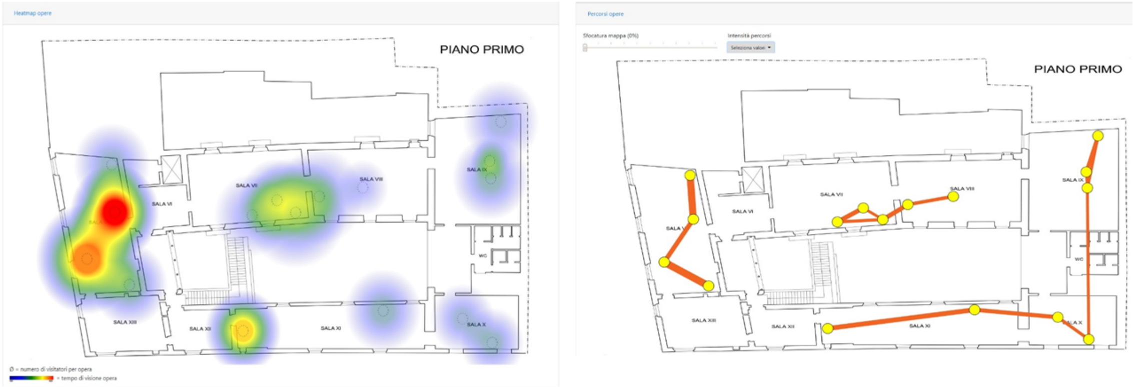
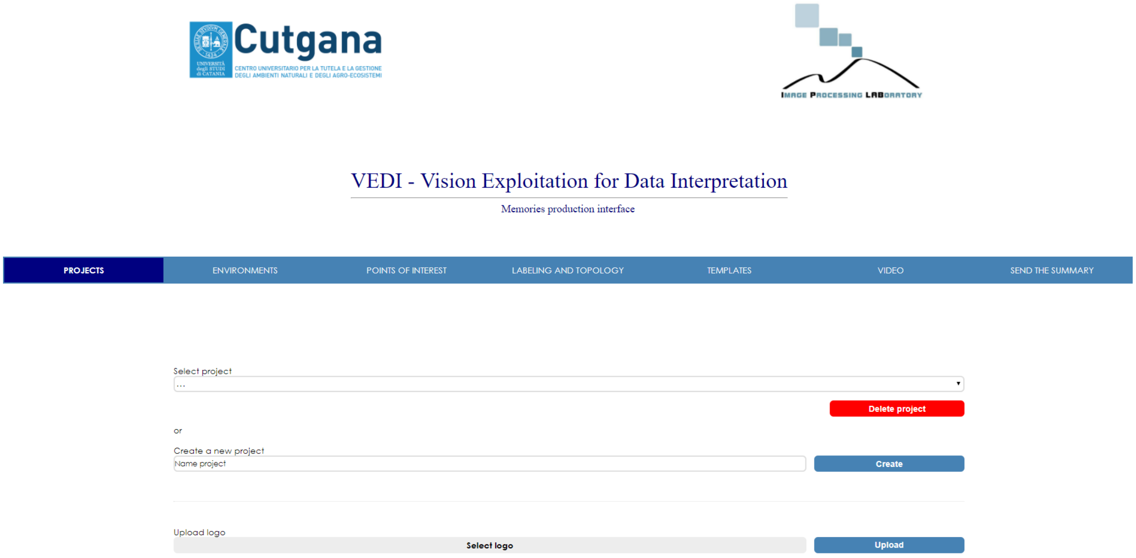


Figure 3.45: An example of data visualization, where a heat map is used to demonstrate the behavior of the visitors.



Cutgana
CENTRO UNIVERSITARIO PER LA TUTELA E LA GESTIONE DEGLI AMBIENTI NATURALI E DEGLI AGRO-ECOSISTEMI

IMAGE PROCESSING LABORATORY

VEDI - Vision Exploitation for Data Interpretation

Memories production interface

PROJECTS | ENVIRONMENTS | POINTS OF INTEREST | LABELING AND TOPOLOGY | TEMPLATES | VIDEO | SEND THE SUMMARY

Select project
...

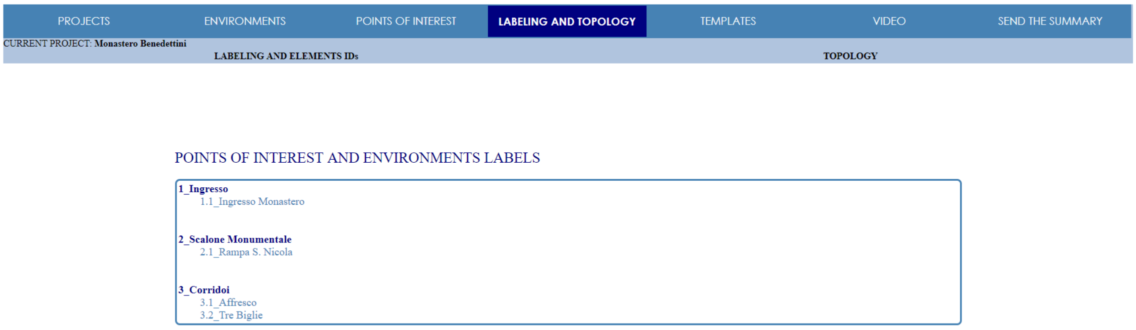
Delete project

or

Create a new project
Name project
Create

Upload logo
Select logo
Upload

Figure 3.46: Management interface.



PROJECTS | ENVIRONMENTS | POINTS OF INTEREST | LABELING AND TOPOLOGY | TEMPLATES | VIDEO | SEND THE SUMMARY

CURRENT PROJECT: **Monastero Benedettini**

LABELING AND ELEMENTS IDs | TOPOLOGY

POINTS OF INTEREST AND ENVIRONMENTS LABELS

1_Ingresso
  1.1_Ingresso Monastero

2_Scalone Monumentale
  2.1_Rampa S. Nicola

3_Corridoi
  3.1_Affresco
  3.2_Tre Biglie

Figure 3.47: The system generates an automatic identifier (ID) for each environment and each point of interest added by the manager.

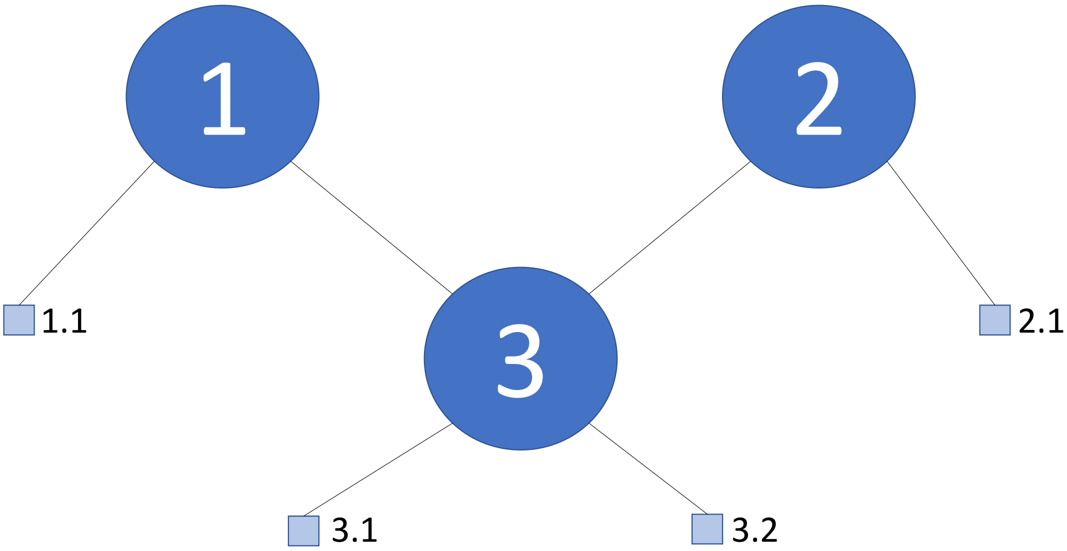
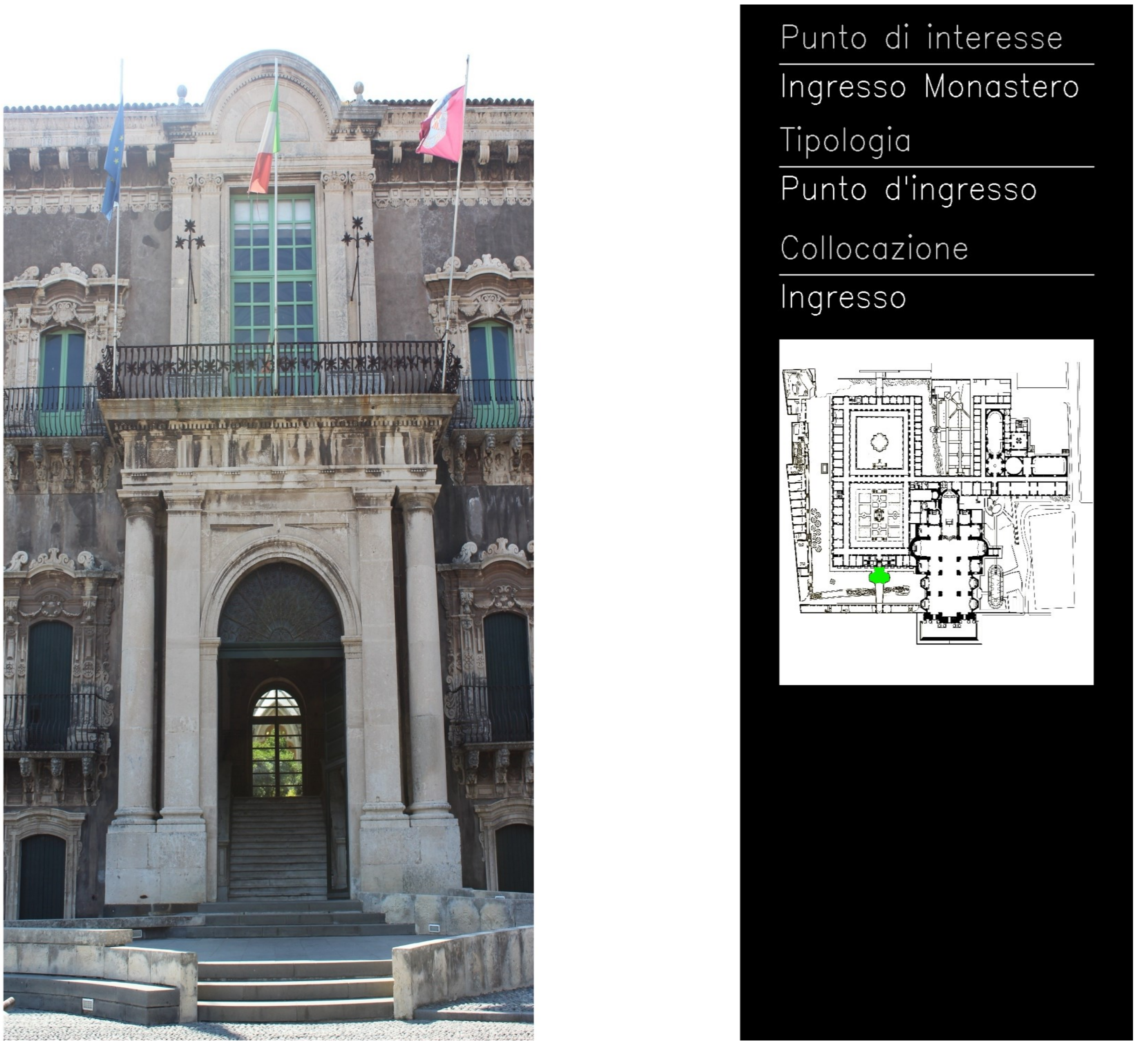


Figure 3.48: An example of topology shown as an undirected graph with 3 environments and 4 points of interest.



Punto di interesse

Ingresso Monastero

Tipologia

Punto d'ingresso

Collocazione

Ingresso

Figure 3.49: Example of template related to an environment.

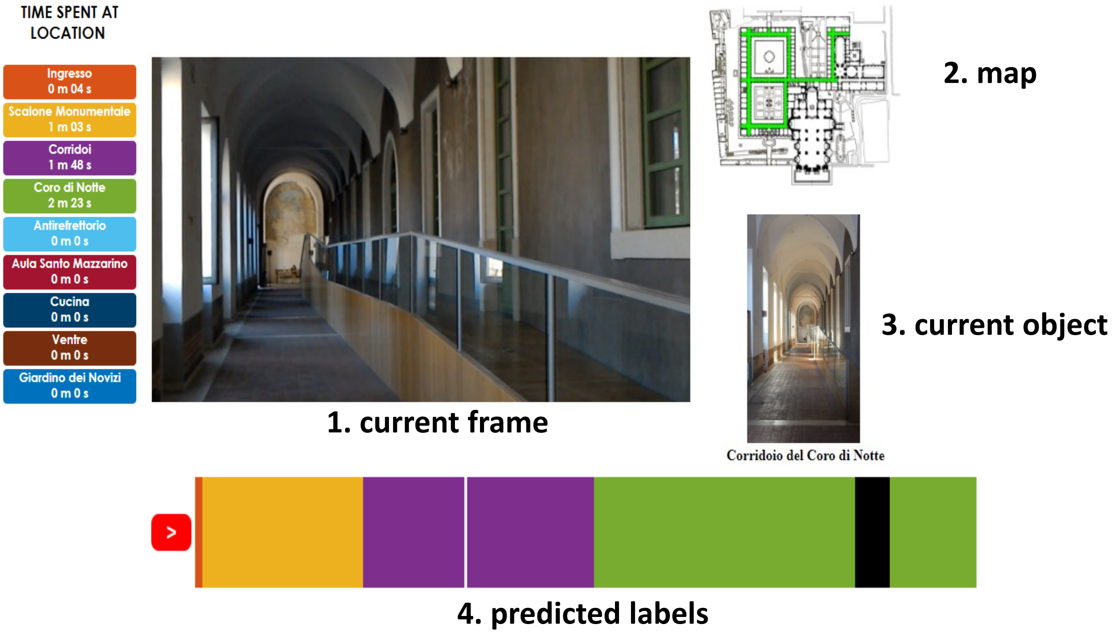


Figure 3.50: The video player is composed by: 1) the current frame of the video; 2) a map that indicates the current location; 3) a pictures of the point of interest observed by the visitor; 4) the predicted location.

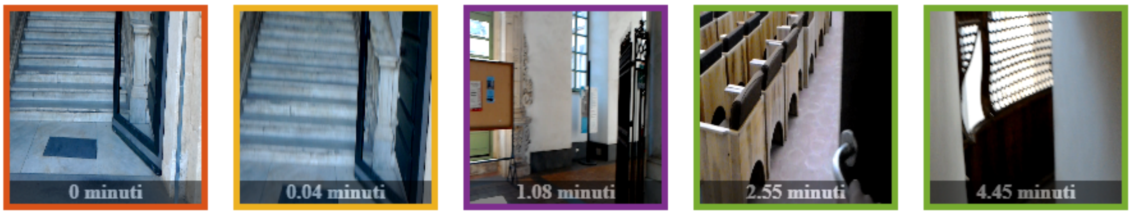


Figure 3.51: Each colored block represents the environment visited by the user in a given video segment. It contains also information on how much time has been spent in that environment.

Chapter 4

Understanding Egocentric Human-Objects Interactions

Understanding human behavior from an egocentric perspective, e.g. using wearable devices, allows to build systems able to improve the safety of workers in a factory or provide assistance to visitors in a museum as discussed in Section ??. In this section, we focus on industrial scenarios, where recognizing human-object interactions can be useful to prevent safety hazards, implement energy saving policies and issue notifications about actions that may be missed in a production pipeline.

We firstly analyze the relations between the concepts of *action* and *interaction* in Section 4.2. Section 4.3 describes MECCANO, the proposed dataset of egocentric videos to study human-object interactions in industrial-like settings and its annotations. In Section 4.4, we report a benchmark aimed to study egocentric human-object interactions in industrial-like domains which shows that the current state-of-the-art approaches achieve limited performance on this challenging dataset.

4.1 Workers Safety in Industrial Domain

Being able to analyze human behavior from egocentric observations has many potential applications related to the recent development of wearable devices [158, 159, 160] which range from improving the personal safety of workers in a factory [161]. Localizing the workers during a fire and assisting them to reach the nearest fire-extinguisher could be an application of a first person system able to guarantee the workers safety. The object detection and recognition problem should be taken into account by an intelligent system in the industrial domain for monitoring the use of

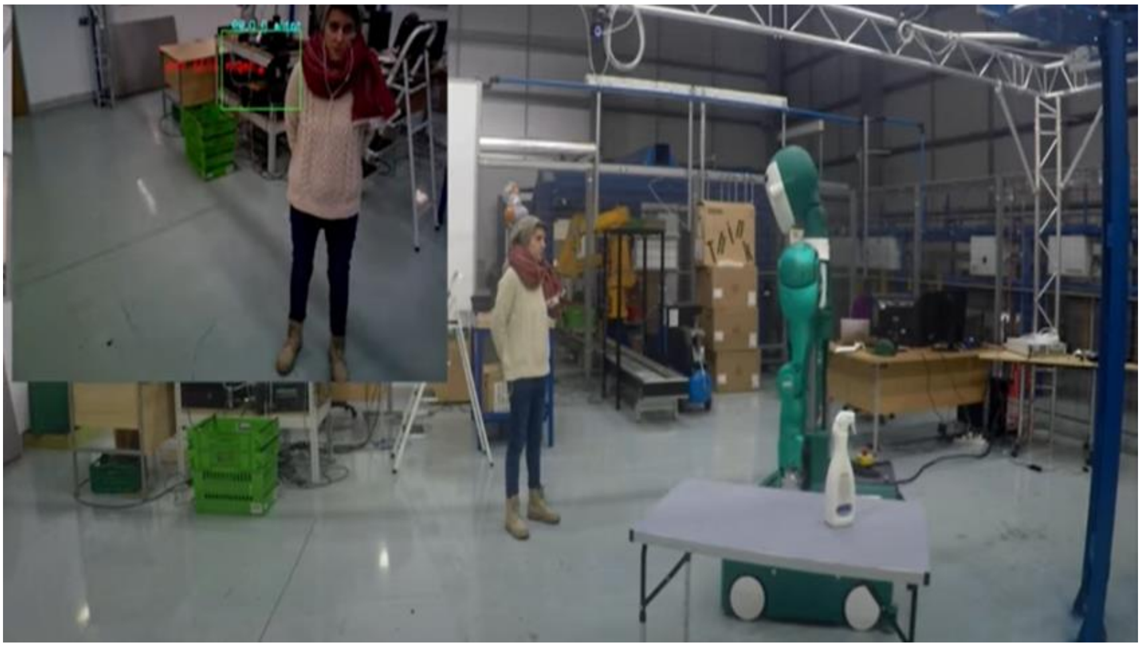


Figure 4.1: Example of a system which assist with a robot the worker.

machines to carry out calibration operations over time and to consider the wear of the objects for maintenance. With the rapid growth of interest in wearable devices in industrial scenarios, recognizing human-object interactions can be useful to prevent safety hazards, implement energy saving policies and issue notifications about actions that may be missed in a production pipeline [162]. Recognizing the human-object interaction in this domain could allow an intelligent system to suggest to the operator how to use a specific machine or object observing a video provided by the wearable device. Recognizing the human-pose and the objects in the surrounding environment, can allow a robot-assistant to give the object the human would take [163] (see Figure 4.1).

In this thesis, we focus on the Egocentric Human-Object Interaction (EHOI) detection considering the industrial domain. The EHOI task has not been previously studied in industrial environments such as factories, building sites, mechanical workshops, etc. This is mainly due to the fact that data acquisition in industrial domains is difficult because of privacy issues and the need to protect industrial secrets. We present the first egocentric dataset (MECCANO) which was acquired in the industrial domain and annotated explicitly to study the EHOI problem (Section 4.4).

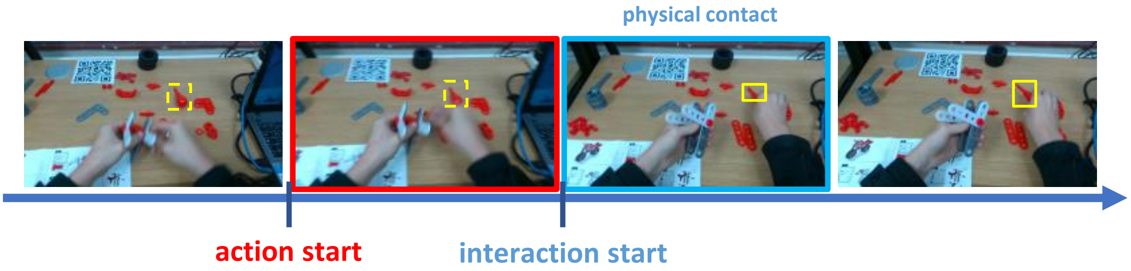


Figure 4.2: Example of relation between action and interaction.

4.2 Action-Interaction Relations

In this section we describe the difference between the *action* and *interaction* concepts which are very related to each other despite different. In the literature, the two concepts are often used interchangeably, specifically when tasks related to *action* recognition and anticipation are considered. Indeed, authors often use the two terms (*action/interaction*) referring to the same thing. In general, both concepts are represented by a verb and an object involved, e.g., "take a screwdriver". We provide definitions which aim to distinguish the two concepts and show that this distinction provides a well reasoned framework to predict human intent.

We assume that, given a verb, a relation exists between the corresponding action and interaction. In particular, we argue that the interaction is strongly related to the contact between the human and one or more objects, whereas actions are more related to motion of the hands of the user and the related objects. For example, if we consider the sentence "take a screwdriver", the action begins when the hand of the human starts to move towards the target object which is the screwdriver (see Figure 4.2). In this phase, the interaction did not start because there was not a physical contact between the hand and the object. When the hand touches the target object (contact) the interaction begins due to the physical contact between the hand and the object, as well as the action is still on-going (see Figure 4.2). When the target object has been taken the action ends, but the physical contact has not been broken, hence, the interaction is not over yet. When the human puts down the object, the interaction will be concluded. This is only one example of relation between action and interaction referred to the verb "take".

Let A_s and A_e be the start and end time of an action and let I_s and I_e denote when the related interaction begins and ends. We found 9 different relations between the two concepts as shown in the Figure 4.3. For each temporal relation, we show the related verbs and the verbs present in the MECCANO dataset if any. Some verbs can be placed simultaneously in different relations due to the fact that a human can perform the same action in different ways. Note that, if there is not a physical contact between the human and the object, the interaction does not exist (e.g. when people run or walk).

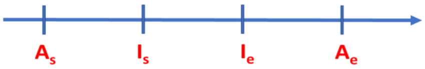
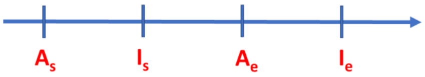
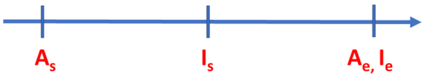
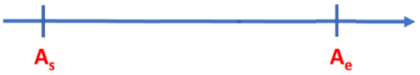
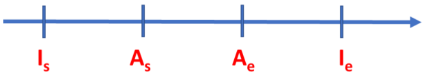
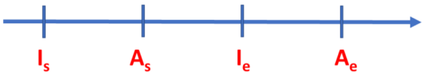
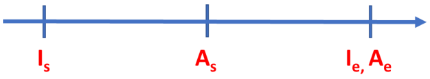
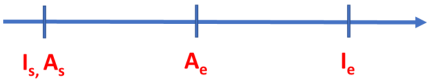
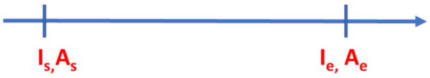
Relation	Verbs	MECCANO verbs
	• pat • hit • kick	//
	• pick up	• take • fit • align • plug • pull
	• close • open • turn on • press • push	• browse
	• walk • jump • run	//
	• wring out • wash • cut • mix	• pull
	• Throw • leave • place	• put
	• move	• browse
	• twist • rip	• screw • unscrew • tighten • loosen
	• stretch • knead • write • watch	• check

Figure 4.3: Relations between action and interaction respect the different verbs.

4.3 The MECCANO Dataset

In this Section, we describe the MECCANO dataset, which is the first dataset of egocentric videos composed of multimodal data related to the industrial-like domain. The multimodality is characterized by the gaze signal, depth maps and RGB videos acquired simultaneously with three different devices.

4.3.1 Data Collection

The MECCANO dataset [164] has been acquired in an industrial-like scenario in which subjects built a toy model of a motorbike (see Figure 4.4). The motorbike is composed of 49 components with different shapes and sizes belonging to 19 classes. In our settings, we have grouped two types of components which are similar in their appearance and have similar roles in the assembly process. Figure 4.5 illustrates the two groups. Specifically, we grouped A054 and A051 under the "screw" class. These two types of components only differ in their lengths. We also grouped A053, A057 and A077 under the "washers" class. Note that these components only differ in the radius of their holes and in their thickness. As a result, we have 20 object classes in total: 16 classes are related to the 49 motorbike components, whereas the others are associated to the two tools, to the instruction booklet and to a partial model class, which indicates a set of components assembled together to form a part of the model (see Figure 4.6). Note that multiple instances of each class are necessary to build the model. In addition, 2 tools, a *screwdriver* and a *wrench*, are available to facilitate the assembly of the toy model. The subjects can follow the instruction booklet while building the toy model.

For the data collection, the 49 components related to the 16 considered classes, the 2 tools and the instruction booklet have been placed on a table to simulate an industrial-like environment. Objects of the same component class have been grouped and placed in a heap, and heaps have been placed randomly (see Figure 4.7).

Other objects not related to the toy model were present in the scene (they constitute clutter background). We have considered two types of tables: a light-colored table and a dark one. The dataset has been acquired by 20 different subjects in 2 countries (Italy and United Kingdom) between May 2019 and January 2020. Participants were from 8 different nationalities with ages between 18 and 55. Figure 4.8

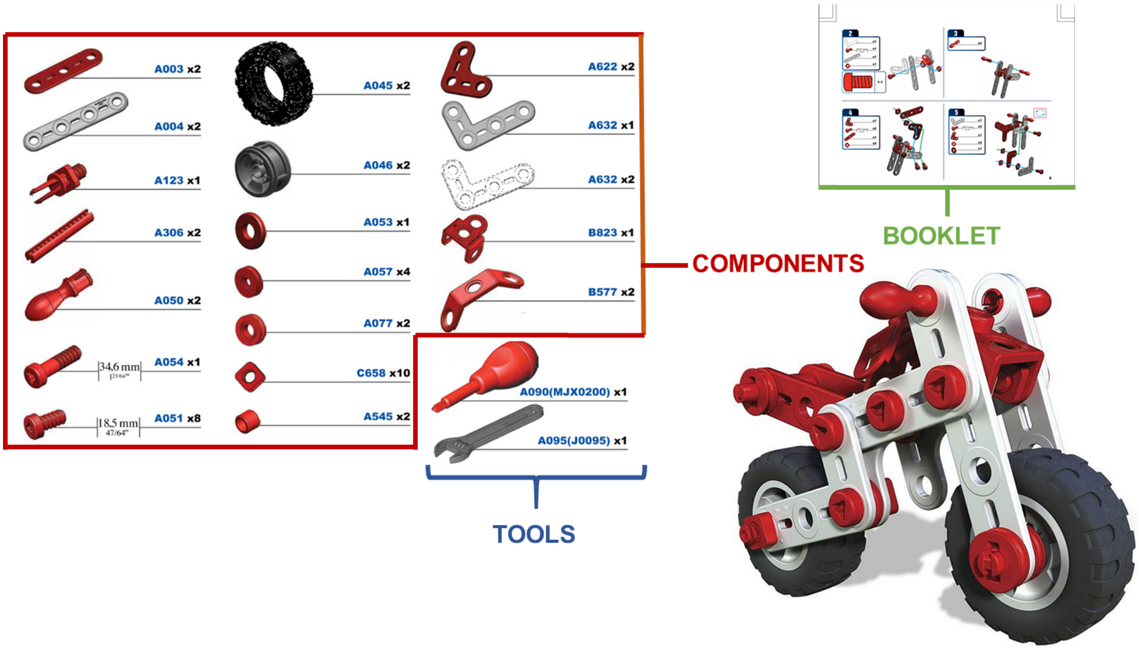


Figure 4.4: Toy model built by subjects interacting with 2 tools, 49 components and the instructions booklet. Better seen on screen.

reports some statistics about the participants. We asked participants to sit and build the model of the motorbike. No other particular instruction was given to the participants, who were free to use all the objects placed on the table as well as the instruction booklet. Some examples of the captured data are reported in Figure 4.7.

The dataset has been acquired using a custom headset (see Figure 4.9) which was worn by participatns for acquisition purposes. The headset was composed of an Intel RealSense SR300[1], a GoPro Hero4[2] and by a Pupils Core[3] device.

The headset was adjusted to control the point of view of the camera with respect to the different heights and postures of the participants in order to have the hands located approximately in the middle of the scene to be acquired.

For each participant, we acquired two RGB streams from the RealSense and GoPro devices, the depth signal from the depth sensor of the RealSense and the gaze signal through the Pupils Core device (see Figure 4.10).

[1] https://ark.intel.com/content/www/it/it/ark/products/92329/intel-realsense-camera-sr300.html

[2] https://gopro.com/it/it/update/hero4

[3] https://pupil-labs.com/

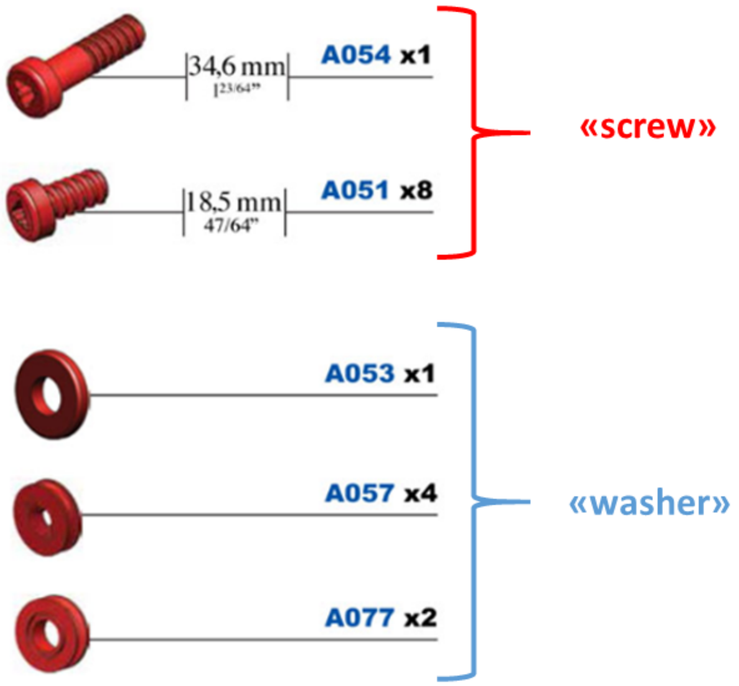


Figure 4.5: Grouped pieces belonging to *screw* and *washer* classes.

The RGB videos acquired with the RealSense device were recorded at a resolution of 1920x1080 pixels. Depth videos were acquired with a resolution of 640x480 pixels. Both videos have a framerate of 12fps. GoPro videos have been acquired with a resolution of 1920x1080 and a framerate of 30fps. Note that the GoPro device has larger field of view respect to the RealSense device (see Figure 4.10). Finally, we acquired the gaze signal with the Pupils Core device with a frequency of 200Hz. To acquire the Real Sense and Pupils Core signals we used the Pupils Capture software[4] which allows to acquire simultaneously with the signals coming from the two devices. To temporally align the GoPro videos with the other signals, participants started each acquisition clapping their hands, which were recorded by both cameras. Each video corresponds to a complete assembly of the toy model starting from the 49 pieces placed on the table. The average duration of the captured videos is 21.14min, with the longest one being 35.45min and the shortest one being 9.23min.

[4] https://pupil-labs.com/products/core/

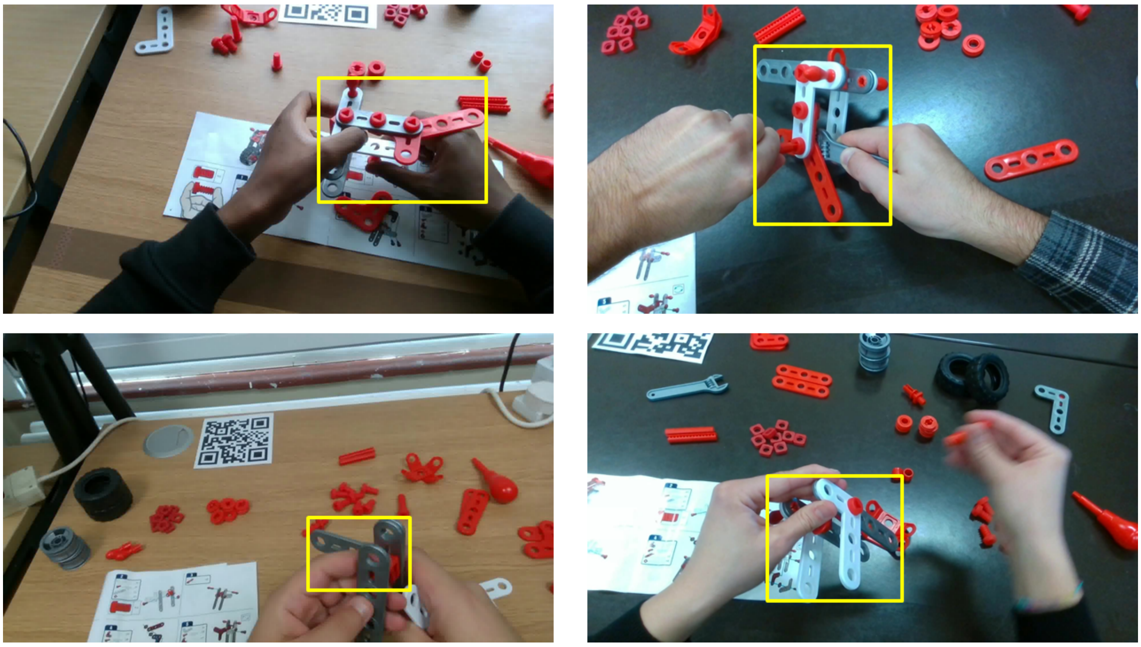


Figure 4.6: Examples of objects belonging to the partial model class.

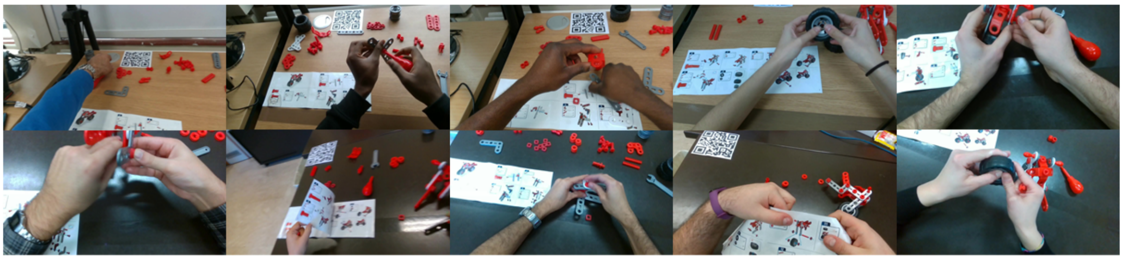


Figure 4.7: Examples of data acquired by the 20 different participants in two countries (Italy, United Kingdom).

4.3.2 Data Annotation

We annotated the MECCANO dataset in two stages. In the first stage, we temporally annotated the occurrences of all human-object interactions indicating their start and end times, as well as a verb describing the interaction. In the second stage, we annotated the *active objects* with bounding boxes for each temporal segment.

Stage 1: Temporal Annotations We considered 12 different verbs which describe the interactions performed by the participants: *take, put, check, browse, plug, pull, align, screw, unscrew, tighten, loosen* and *fit*. As shown in Figure 4.11, the distribution of the verb classes of the labeled samples in our dataset follows a

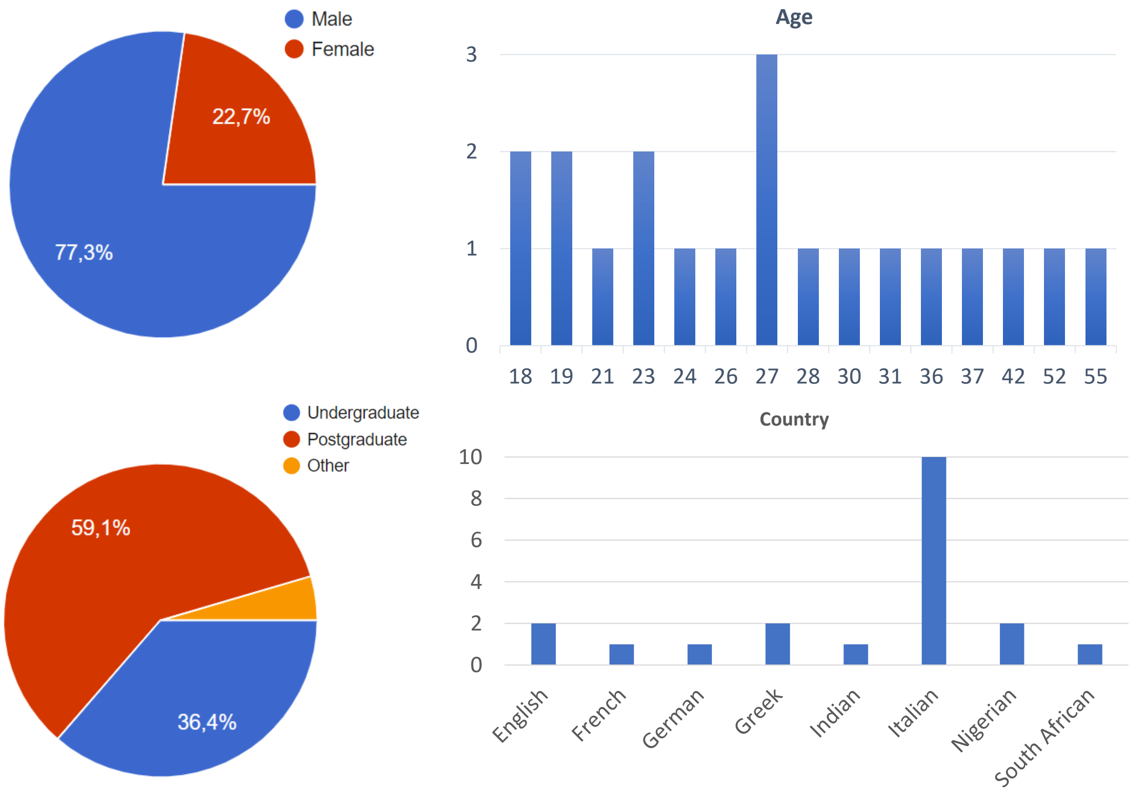


Figure 4.8: Statistics of the 20 participants.

long-tail distribution, which suggests that the taxonomy captures the complexity of the considered scenario. Figure 4.12 reports the percentage of the temporally annotated instances belonging to the 12 verb classes. Each temporal segment has been annotated considering the contact (i.e., between the hand and the object or between the objects) as the start time of the segment and the end of the correspondent *action* as the end time of the temporal segment. We used the ELAN Annotation tool [165] to annotate a temporal segment around each instance of an action. Each segment has been associated to the verb which best described the contained action.

Since a participant can perform multiple actions simultaneously, we allowed the annotated segments to overlap (see Figure 4.13). In particular, in the MECCANO dataset there are 1401 segments (15.82 %) which overlap with at least another segment. We consider the start time of a segment as the timestamp in which the hand touches an object, changing its state from *passive* to *active*. The only exception is for the verb *check*, in which case the user doesn't need to touch an object to

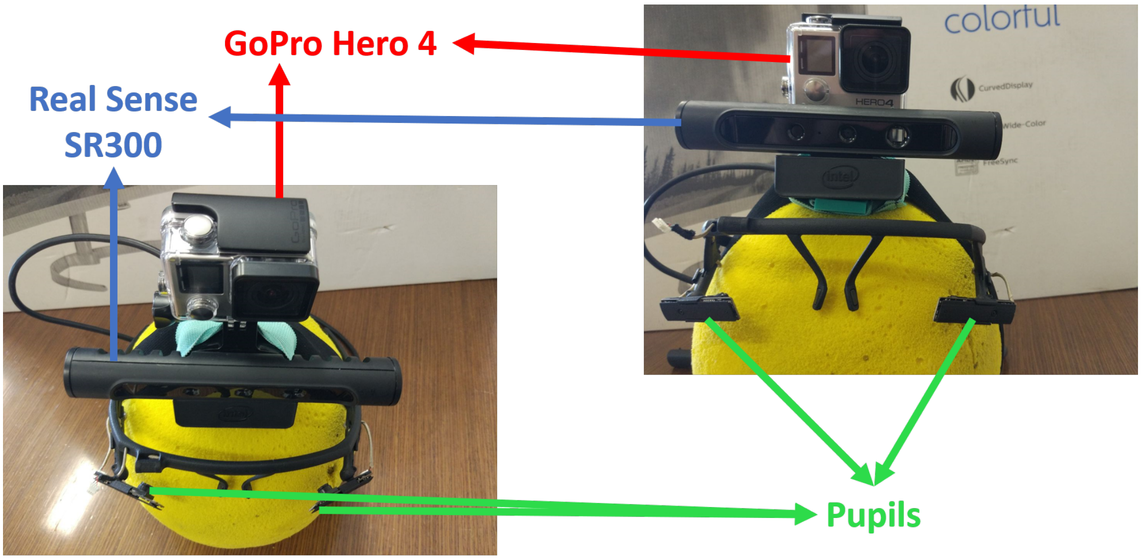


Figure 4.9: The custom headset used to acquire the MECCANO Multimodal dataset.

perform an interaction. In this case, we annotated the start time when it is obvious from the video sequence that the user is looking at the object (see Figure 4.13). With this procedure, we annotated 8857 video segments.

Stage 2: Active Object Bounding Box Annotations We considered 20 object classes which include the 16 classes categorizing the 49 components, the two tools (*screwdriver* and *wrench*), the instructions booklet and a *partial_model* class. The latter object class represents assembled components of the toy model which are not yet complete (e.g., a *screw* and a *bolt* fixed on a *bar* which have not yet been assembled with the rest of the model). Some examples of the *partial_model* class are shown in Figure 4.6. For each temporal segment, we annotated the *active* objects in frames sampled every 0.2 seconds. Each active object annotation consists in a *(class, x, y, w, h)* tuple, where *class* represents the class of the object and *(x, y, w, h)* are the 2D coordinates which define a bounding box around the object in the frame. We annotated multiple objects when they were active simultaneously (see Figure 4.14 - first row). Moreover, if an active object is occluded, even just in a few frames, we annotated it with a *(class, x, y)* tuple, specifying the class of the object and its estimated 2D position. An example of occluded active object annotation is reported in the second row of Figure 4.14.

For the bounding box annotation procedure, we used VGG Image Annotator

(VIA) [166] with a customized project which allowed annotators to select component classes from a dedicated panel showing the thumbnails of each of the 20 object classes to facilitate and speed up the selection of the correct object class. Figure 4.15 reports an example of the customized VIA interface. Moreover, to support annotators and reduce ambiguities, we prepared a document containing a set of fundamental rules for the annotations of *active* objects, where we reported the main definitions (e.g., active object, occluded active object, partial_model) along with visual examples. Figure 4.16 reports an example of such instructions. With this procedure, we labeled a total of 64349 frames.

Action Annotations

Starting from the temporal annotations, we defined 61 action classes. Each action is composed by a verb and one or more objects, for example *"align screwdriver to screw"* in which the verb is *align* and the objects are *screwdriver* and *screw*. Depending on the verb and objects involved in the interaction, each temporal segment has been associated to one of the 61 considered action classes. Figure 4.17 shows the list of the 61 action classes, which follow a long-tail distribution. We analyzed the combinations of our 12 verb classes and 20 object classes to find a compact, yet descriptive set of actions classes. The action class selection process has been performed in two stages. In the first stage, we obtained the distributions of the number of active objects generally occurring with each of the 12 verbs. The distributions are shown in Figure 4.18. For example, the dataset contains 120 instances of "browse" (second row - first column), which systematically involves one single object. Similarly, most of the instance of "take" appear with 1 object, while few instances have $2-3$ objects.

In the second stage, we selected a subset of actions from all combinations of verbs and nouns. Figure 4.19 reports all the action classes obtained from the 12 verbs classes of the MECCANO dataset as discussed in the following.

Let $O = \{o_1, o_2, ..., o_n\}$ and $V = \{v_1, v_2, ..., v_m\}$ be the set of the objects and verb classes respectively. For each verb $v \in V$, we considered all the object classes $o \in O$ involved in one or more temporal segments labeled with verb v. We considered the following rules:

- **Take and put**: We observed that all the objects $o \in O$ occurring with $v = take$ are taken by participants while they build the motorbike. Hence, we first defined 20 action classes as (v, o) pairs (one for each of the available objects). Since subjects can take more than one object at a time, we added an additional "take objects" action class when two or more objects are taken simultaneously. The same behavior has been observed for the verb $v = put$. Hence, we similarly defined 21 action classes related to this verb.

- **Check and browse**: We observed that verbs $v = check$ and $v = browse$ always involve only the object $o = instruction\ booklet$. Hence, we defined the two action classes *check instruction booklet* and *browse instruction booklet.*

- **Fit**: When the verb is $v = fit$, there are systematically two objects involved simultaneously (i.e., $o = rim$ and $o = tire$). Hence, we defined the action class *fit rim and tire.*

- **Loosen**: We observed that participants tend to loosen bolts always with the hands. We hence defined the action class *loosen bolt with hands.*

- **Align**: We observed that participants tend to align the screwdriver tool with the screw before starting to screw, as well as the wrench tool with the bolt before tightening it. Participants also tended to align objects to be assembled to each other. From these observations, we defined three action classes related to the verb $v = align$: *align screwdriver to screw*, *align wrench to bolt* and *align objects.*

- **Plug**: We found three main uses of verb $v = plug$ related to the objects $o = screw$, $o = rod$ and $o = handlebar$. Hence, we defined three action classes: *plug screw*, *plug rod* and *plug handlebar.*

- **Pull**: Similar observations apply to verb $v = pull$. Hence we defined three action classes involving "pull": *pull screw*, *pull rod* and *pull partial model.*

- **Screw and unscrew**: The main object involved in actions characterized by the verbs $v = screw$ and $v = unscrew$ is $o = screw$. Additionally, the screw or unscrew action can be performed with a screwdriver or with hands. Hence, we defined four action classes *screw screw with screwdriver, screw screw with hands, unscrew screw with screwdriver* and *unscrew screw with hands.*

- **Tighten**: Similar observation holds for the verb $v = tighten$, the object $o = bolt$ and the tool $o = wrench$. We hence defined the following two action classes: *tighten bolt with wrench* and *tighten bolt with hands.*

In total, we obtained 61 action classes composing the MECCANO dataset.

EHOI Annotations

The HOI detection task consists in detecting the occurrence of human-object interactions, localizing both the humans taking part in the action and the interacted objects. HOI detection also aims to understand the relationships between humans and objects, which is usually described with a verb. Possible examples of HOIs are *"wash the plates"* or *"open the door"*. HOI detection models mostly consider one single object involved in the interaction [8, 15, 9, 16, 11]. Hence, an interaction is defined as a triplet in the form <*human, verb, object*>, where the human is the subject of the action specified by a verb and an object.

We define Egocentric Human-Object Interaction (EHOI) detection as the task of producing <verb, objects> pairs describing the interaction observed from the egocentric point of view. Note that in EHOI, the human interacting with the objects is always the camera wearer, while one or more objects can be involved simultaneously in the interaction.

Let $O = \{o_1, o_2, ..., o_n\}$ and $V = \{v_1, v_2, ..., v_m\}$ be the sets of objects and verbs respectively. We define an Egocentric Human-Object Interaction e as:

$$e = (v_h, \{o_1, o_2, ..., o_i\}) \tag{4.1}$$

where $v_h \in V$ is the verb characterizing the interaction and $(o_1, o_2, ..., o_i) \subseteq O$ represent the active objects involved in the interaction. Given the previous definition, we considered all the observed combinations of verbs and objects to represent EHOIs performed by the participants during the acquisition (see examples in Figure 4.20). Each EHOI annotation is hence composed of a verb annotation and the *active* object bounding boxes. The MECCANO dataset is the first dataset of egocentric videos explicitly annotated for the EHOI detection task.

Next Active Object Annotations

We annotated MECCANO with a set of annotations useful to tackle the problem of *Next Active Objects* prediction. For each human-object interaction, we annotated the objects which will be *active objects* in the frames preceding the interaction (i.e., contact frame). Moreover, we annotated the hands with bounding boxes over the frames belonging to the interaction and in the frames which represent the past of the interaction.

For each temporal segment which represents a human-object interaction, we considered the frame when the interaction starts which corresponds to the start frame of the temporal segment (see Section 4.3.2). We sampled frames every 0.2 seconds going back up to 3 seconds before the beginning of the temporal segment, or less if there is an overlap with a previous segment. Indeed, not all interactions have past frames. An example of the sampling procedure related to the interaction "take bolt" is shown in Figure 4.21.

Figure 4.22 shows the comparison between the number of interactions present in the MECCANO dataset with respect to the number of interactions which include labeled past frames. With this sampling procedure, we obtained labels in past frames for the 75.66% (6656) of the total number of interactions (8857) present in the dataset.

Considering the past frames of an interaction, each next-active object annotation consists in a *(class, x, y, w, h)* tuple, where "class" represents the class of the object which will be active and *(x, y, w, h)* defines a bounding box around the considered object. If an object is going to be taken from a pile, then the pile itself is labeled. Note that a pile of objects is composed only by objects of the same type. We labeled the pile because we assume that before a human-object interaction occurs it is not feasible to infer which object of the pile will be active (see Figure 4.23). If the object is occluded, we annotated it with a *(class, x, y)* tuple specifying the class of the object and its estimated 2D position. With this procedure, we labeled a total of 48024 frames with 74127 bounding boxes.

As shown in Figure 4.24, the distribution of bounding boxes over all object classes follows a long-tail distribution, which highlights the complexity of the considered scenario. Moreover, we reported how many bounding boxes we annotated for each object class considering the three splits (Training, Validation and Test) of the dataset.

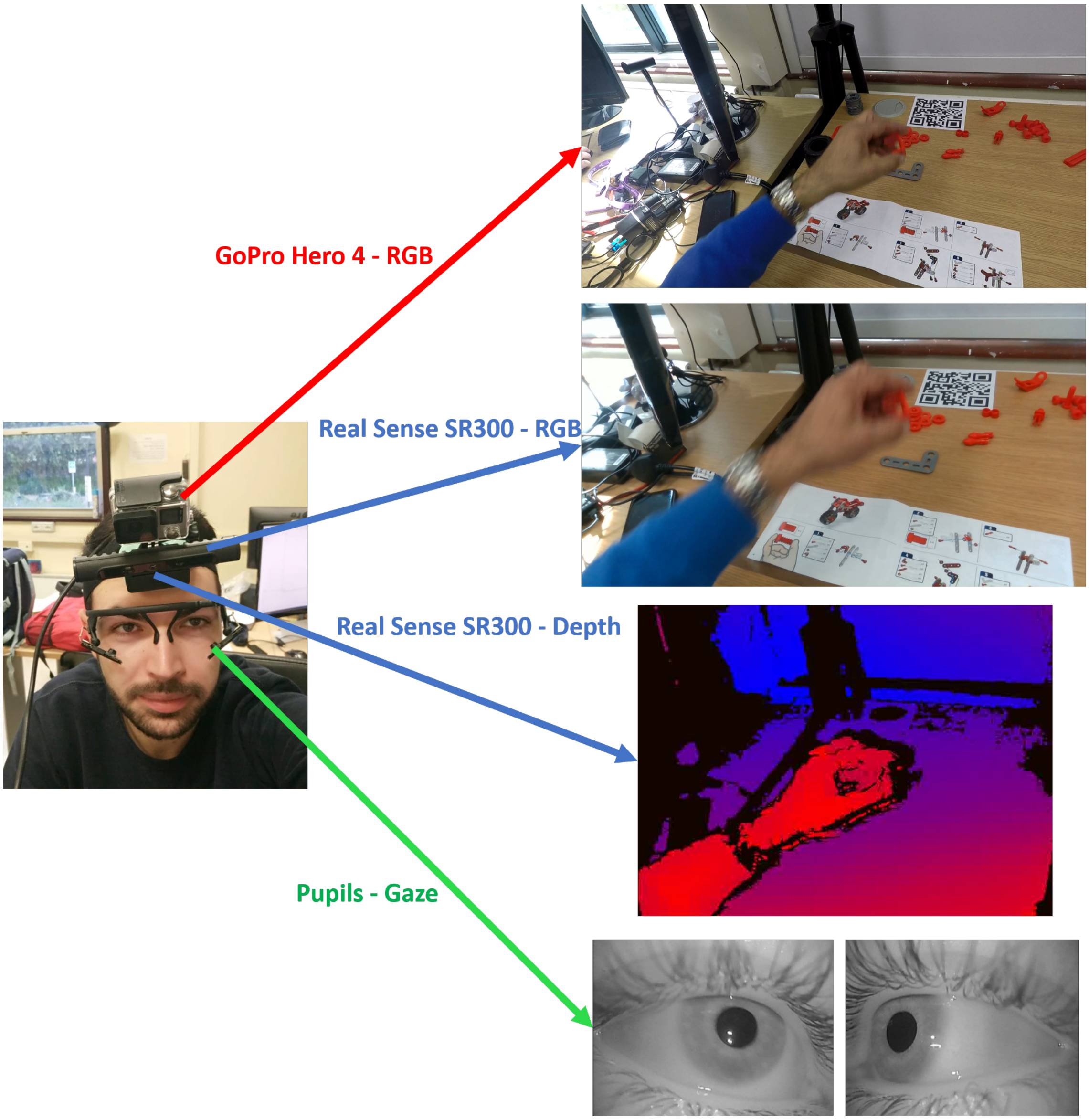


Figure 4.10: The multimodal signals considered in the MECCANO Multimodal Dataset.

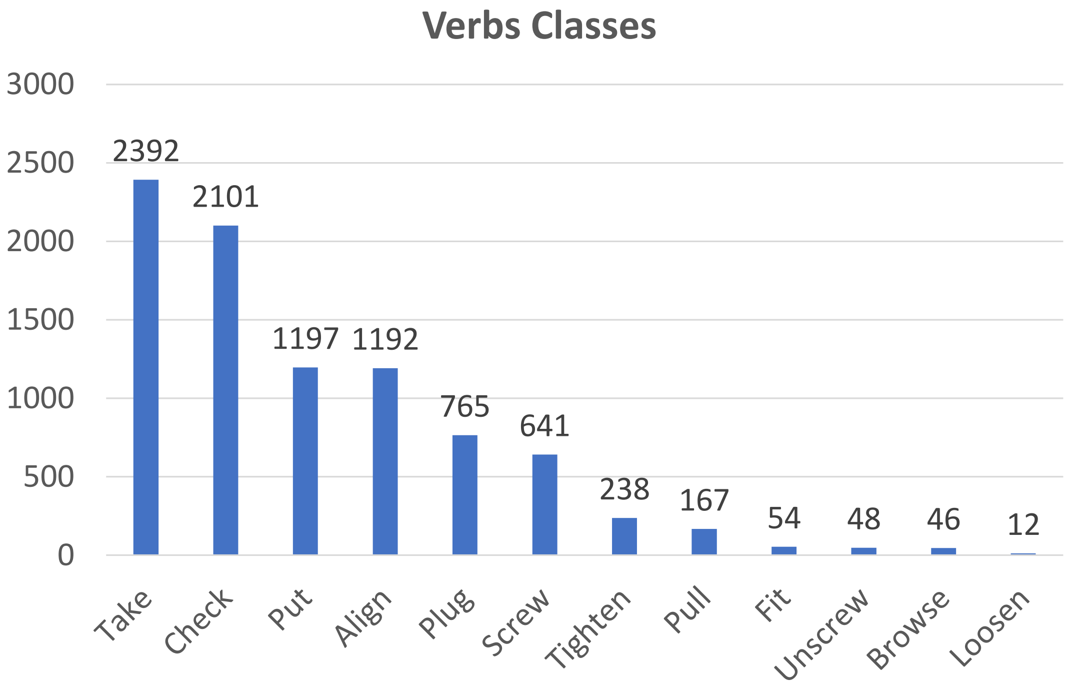


Figure 4.11: Long-tail distribution of verbs classes.

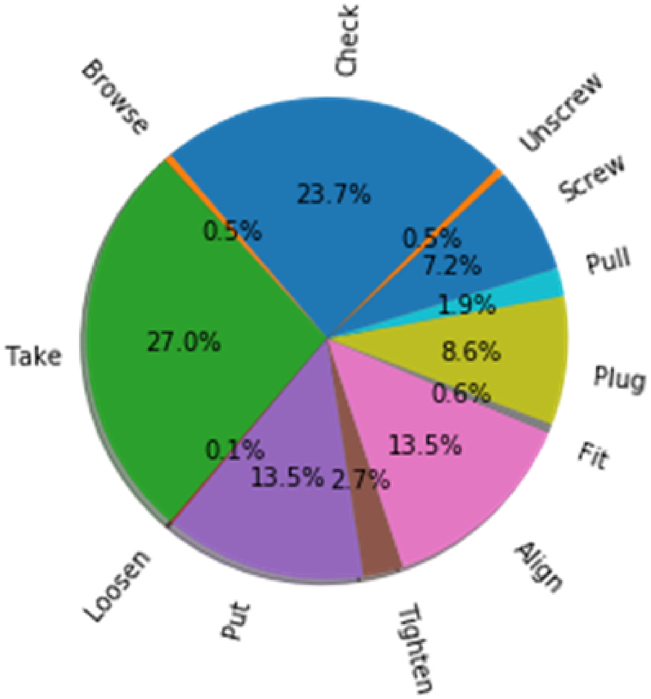


Figure 4.12: Fractions of instances of each verb in the MECCANO dataset.

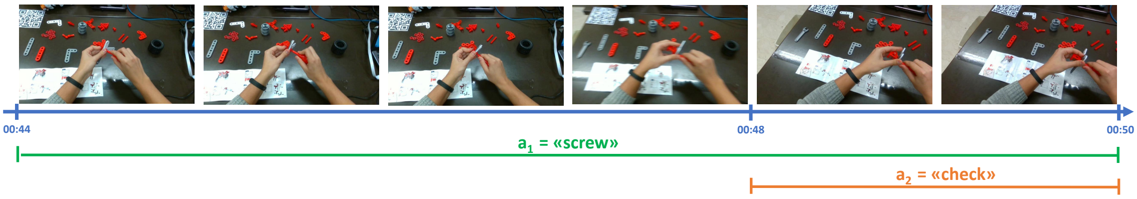


Figure 4.13: Example of two overlapping temporal annotations along with the associated verbs.

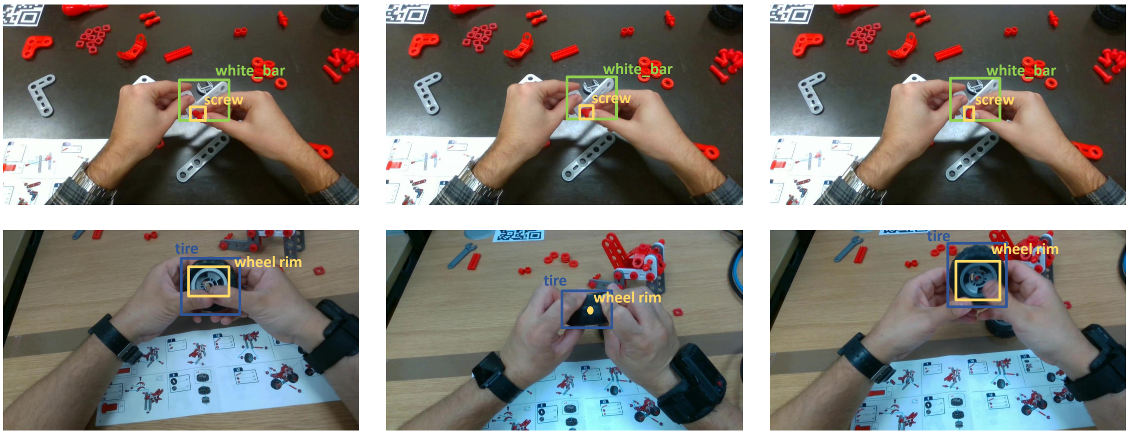
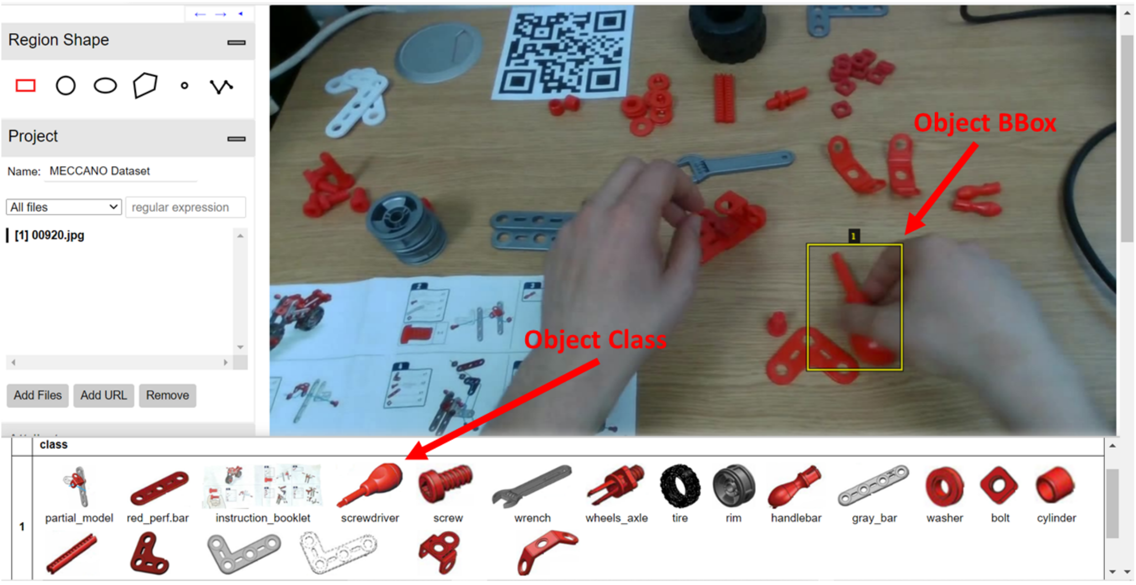
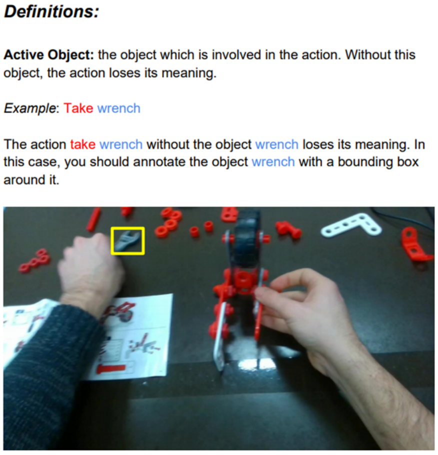


Figure 4.14: Example of bounding box annotations for *active* objects (first row) and occluded *active* objects (second row).


Name: MECCANO Dataset

[1] 00920.jpg

Figure 4.15: Customized VIA project to support the labeling of active objects. Annotators were presented with a panel which allowed them to identify object classes through their thumbnails.



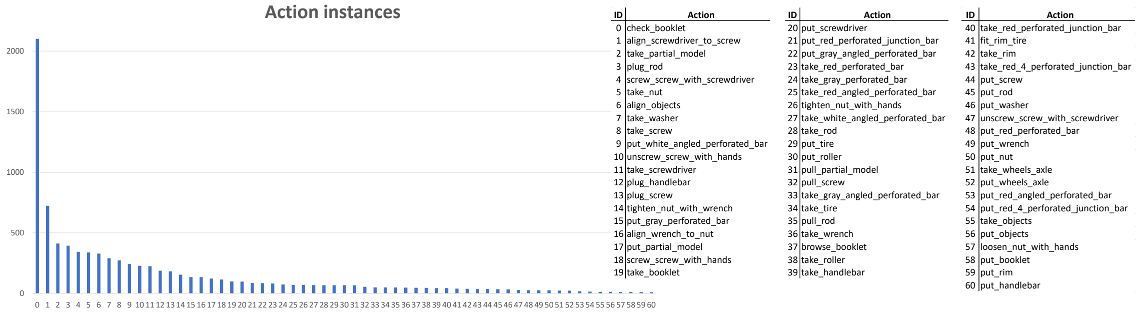
Definitions:

Active Object: the object which is involved in the action. Without this object, the action loses its meaning.

Example: Take wrench

The action take wrench without the object wrench loses its meaning. In this case, you should annotate the object wrench with a bounding box around it.

Figure 4.16: *Active* object definition given to the labelers for the *active* object bounding box annotation stage.



ID	Action	ID	Action	ID	Action
0	check_booklet	20	put_screwdriver	40	take_red_perforated_junction_bar
1	align_screwdriver_to_screw	21	put_red_perforated_junction_bar	41	fit_rim_tire
2	take_partial_model	22	put_gray_angled_perforated_bar	42	take_rim
3	plug_rod	23	take_red_perforated_bar	43	take_red_4_perforated_junction_bar
4	screw_screw_with_screwdriver	24	take_gray_perforated_bar	44	put_screw
5	take_nut	25	take_red_angled_perforated_bar	45	put_rod
6	align_objects	26	tighten_nut_with_hands	46	put_washer
7	take_washer	27	take_white_angled_perforated_bar	47	unscrew_screw_with_screwdriver
8	take_screw	28	take_rod	48	put_red_perforated_bar
9	put_white_angled_perforated_bar	29	put_tire	49	put_wrench
10	unscrew_screw_with_hands	30	put_roller	50	put_nut
11	take_screwdriver	31	pull_partial_model	51	take_wheels_axle
12	plug_handlebar	32	pull_screw	52	put_wheels_axle
13	plug_screw	33	take_gray_angled_perforated_bar	53	put_red_angled_perforated_bar
14	tighten_nut_with_wrench	34	take_tire	54	put_red_4_perforated_junction_bar
15	put_gray_perforated_bar	35	pull_rod	55	take_objects
16	align_wrench_to_nut	36	take_wrench	56	put_objects
17	put_partial_model	37	browse_booklet	57	loosen_nut_with_hands
18	screw_screw_with_hands	38	take_roller	58	put_booklet
19	take_booklet	39	take_handlebar	59	put_rim
				60	put_handlebar

Figure 4.17: Distribution of action instances in the MECCANO dataset.

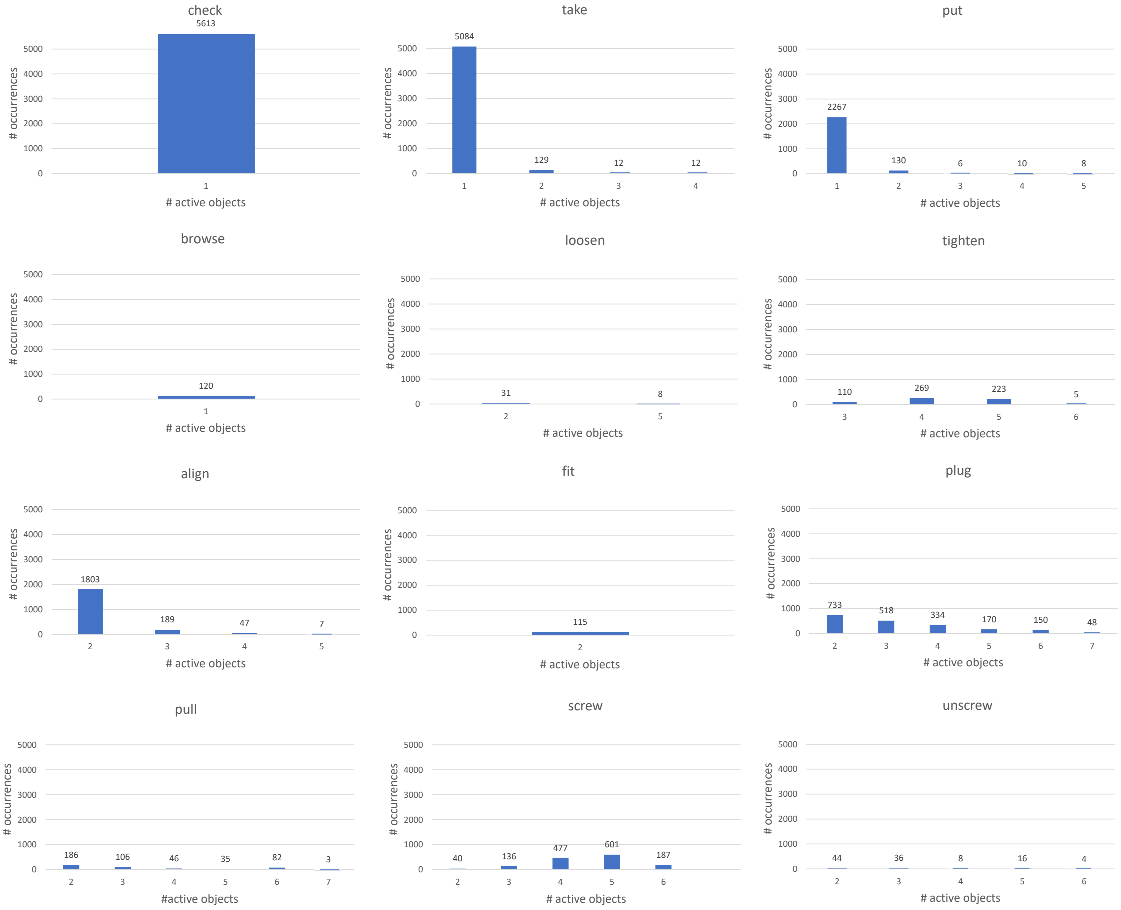


Figure 4.18: Number of objects and occurrences of *active* objects related to each verb.

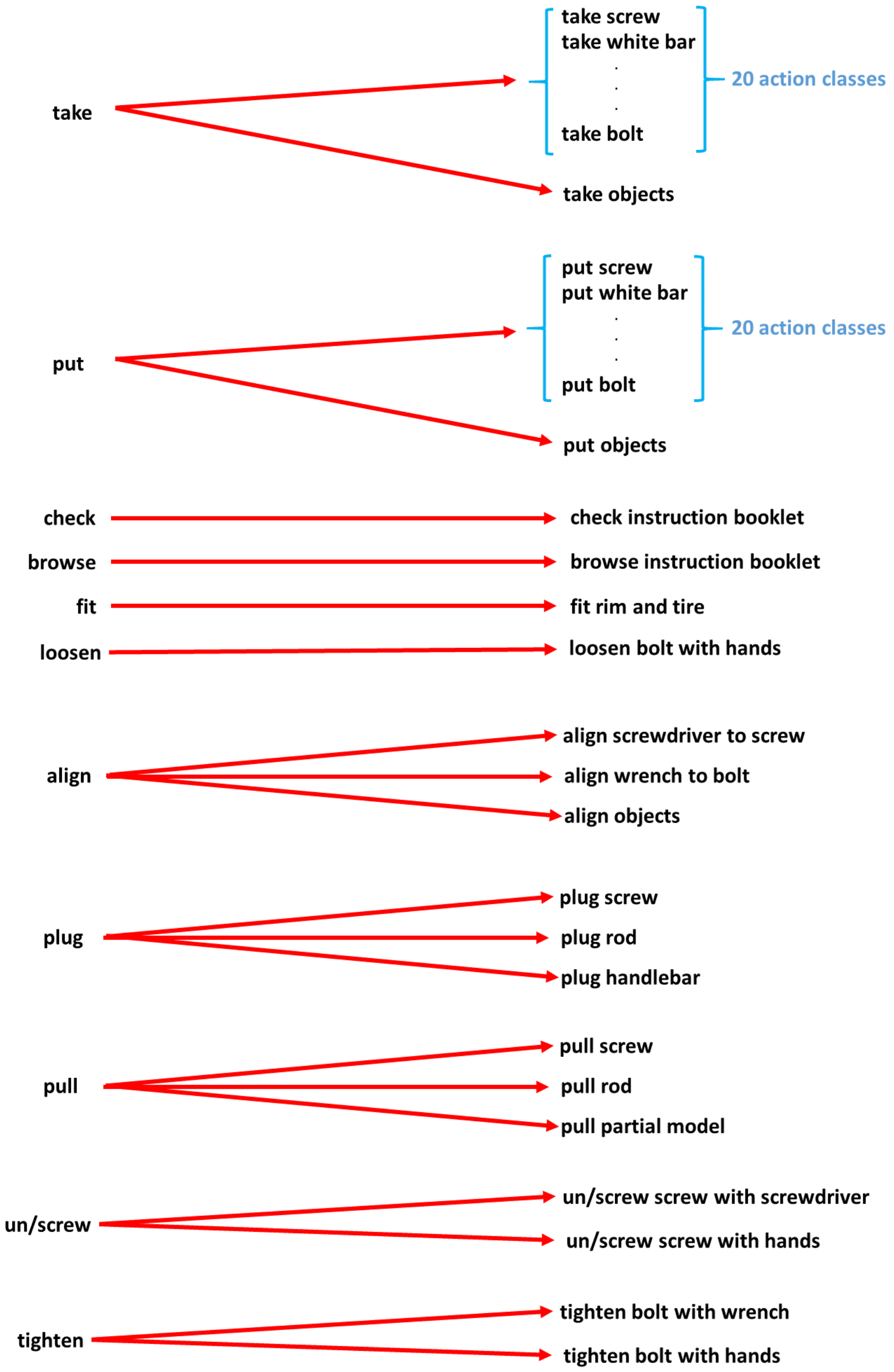


Figure 4.19: 61 action classes definition from the 12 verb classes and the analysis performed observing the participant behavior.

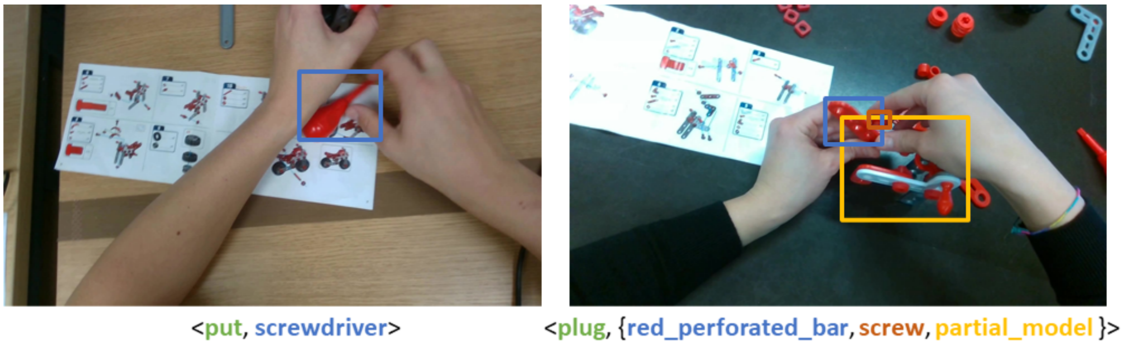


Figure 4.20: Some examples of EHOI.

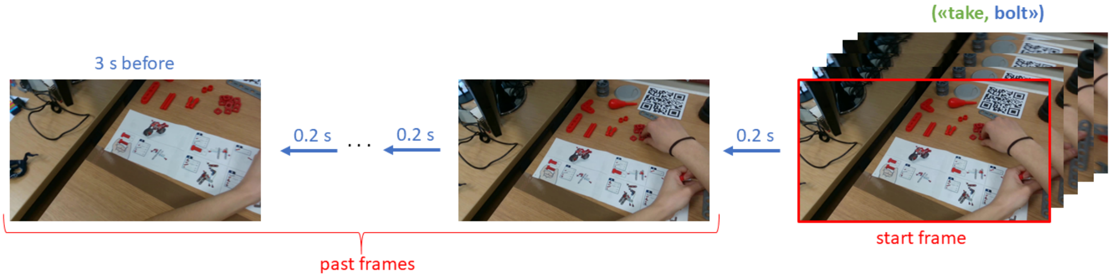
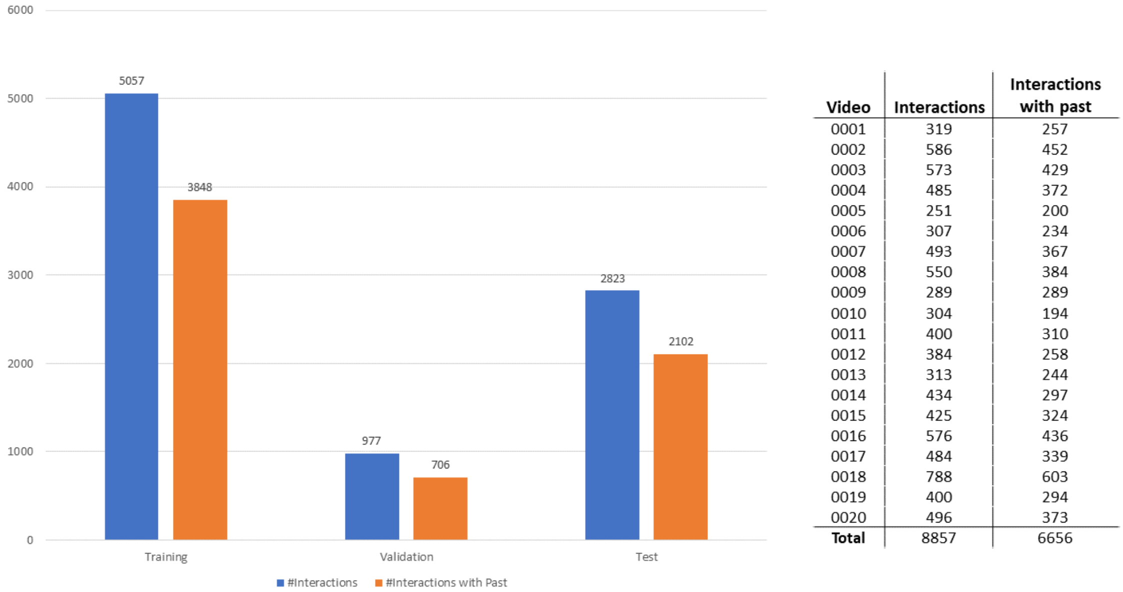


Figure 4.21: The sampling procedure adopted to obtain the past frames for each interaction of the MECCANO Multimodal dataset.



Video	Interactions	Interactions with past
0001	319	257
0002	586	452
0003	573	429
0004	485	372
0005	251	200
0006	307	234
0007	493	367
0008	550	384
0009	289	289
0010	304	194
0011	400	310
0012	384	258
0013	313	244
0014	434	297
0015	425	324
0016	576	436
0017	484	339
0018	788	603
0019	400	294
0020	496	373
Total	8857	6656

Figure 4.22: Comparison between the number of interactions with respect to the number of interactions which have past frames.

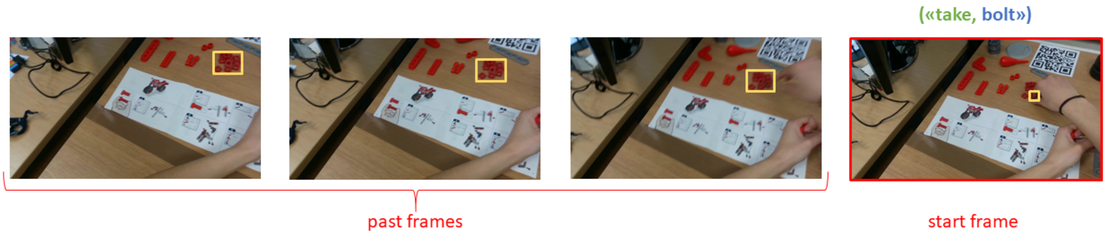
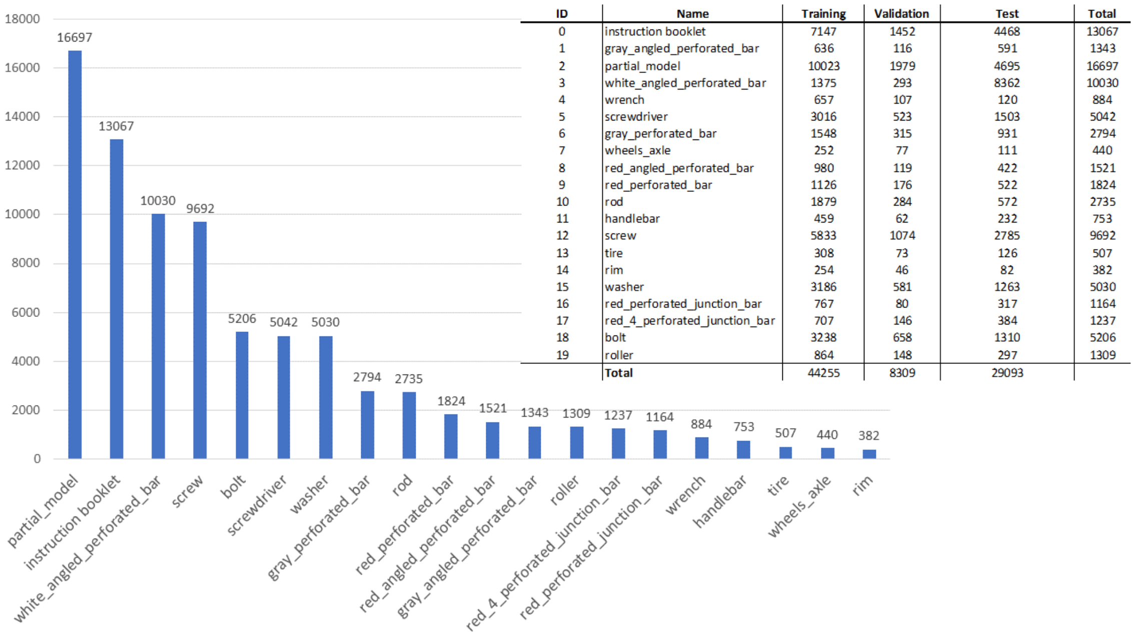


Figure 4.23: Example of next active object annotation where, the object is going to be taken from a pile.



ID	Name	Training	Validation	Test	Total
0	instruction booklet	7147	1452	4468	13067
1	gray_angled_perforated_bar	636	116	591	1343
2	partial_model	10023	1979	4695	16697
3	white_angled_perforated_bar	1375	293	8362	10030
4	wrench	657	107	120	884
5	screwdriver	3016	523	1503	5042
6	gray_perforated_bar	1548	315	931	2794
7	wheels_axle	252	77	111	440
8	red_angled_perforated_bar	980	119	422	1521
9	red_perforated_bar	1126	176	522	1824
10	rod	1879	284	572	2735
11	handlebar	459	62	232	753
12	screw	5833	1074	2785	9692
13	tire	308	73	126	507
14	rim	254	46	82	382
15	washer	3186	581	1263	5030
16	red_perforated_junction_bar	767	80	317	1164
17	red_4_perforated_junction_bar	707	146	384	1237
18	bolt	3238	658	1310	5206
19	roller	864	148	297	1309
	Total	44255	8309	29093	

Figure 4.24: Long-tail distribution of bounding boxes over all object classes.

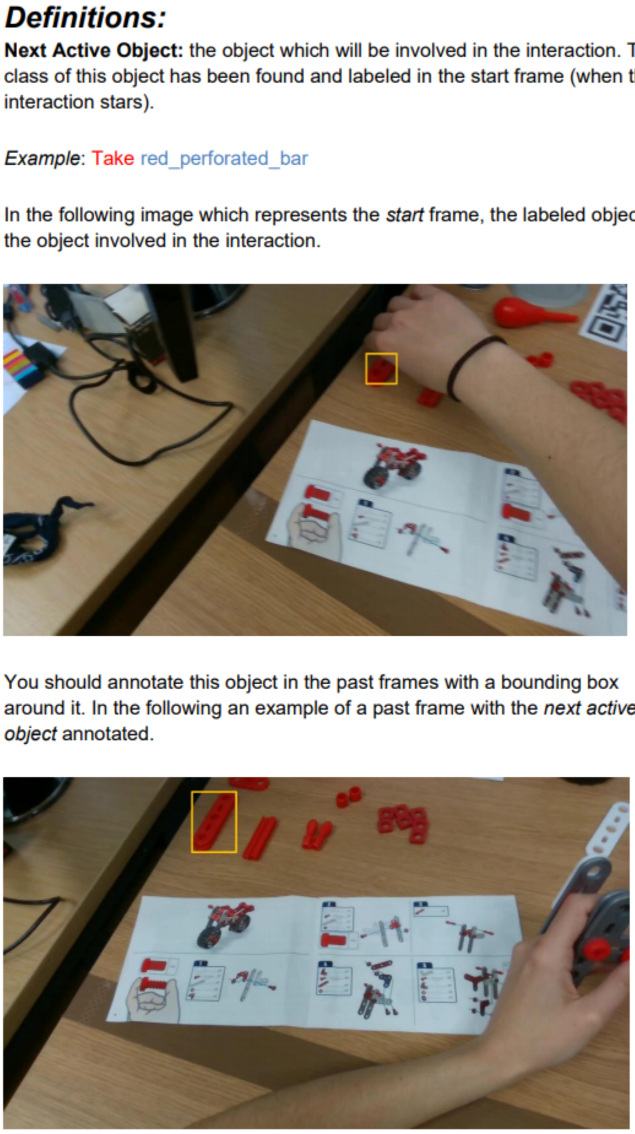
For the annotation phase, we used VGG Image Annotator (VIA) [166] with the customized project described in Section 4.3.2. Moreover, we provided a document to the annotators, containing a set of key rules for the annotations of next active objects, to support annotators and reduce ambiguities. In annotation guidelines, we reported the fundamental definitions (e.g., next active object, next active object in a pile, occluded next active object) showing visual examples (see Figure 4.25).

Hands Annotations

For each interaction, we annotated the hands of the participants with a bounding box on the set of frames belonging to the interaction (i.e. from the start frame to the end frame) and in the past frames preceding the interaction. Each hand annotation consists in a (class, x, y, w, h) tuple, where "class" represents the side of the hand (i.e. left or right) and (x, y, w, h) defines a bounding box around the considered hand as shown in Figure 4.26 and Figure 4.27.

We split this labeling procedure in two stages. Firstly, we processed the frames with the Hand Object Detector described in [112]. This detector infers if an hand is involved in an interaction through the contact with active objects. In particular, the detector predicts the hand location, the side, a contact state, and a box around the object in contact. We considered only the hand location and the side for each of the processed frame. In the second stage, the annotators checked if the predicted bounding boxes and the associated class were precise and correct or if there was a missing hand prediction. If the bounding box was not precise or the class was wrong, they refined the bounding box and corrected the class of the hand. An example of this labeling procedure is shown in Figure 4.27. In the first column, we reported the predictions of the Hand Object Detector. In the second column, the annotators fixed the class errors and refined the bounding box around the hands.

With this procedure, we annotated *89628* frames with *169625* bounding boxes around the hands. Figure 4.28 reports some statistics related to the hand annotations.



Definitions:

Next Active Object: the object which will be involved in the interaction. The class of this object has been found and labeled in the start frame (when the interaction stars).

Example: Take red_perforated_bar

In the following image which represents the *start* frame, the labeled object is the object involved in the interaction.

You should annotate this object in the past frames with a bounding box around it. In the following an example of a past frame with the *next active object* annotated.

Figure 4.25: Next active object definition given to the labelers for the next active object bounding box annotation stage.

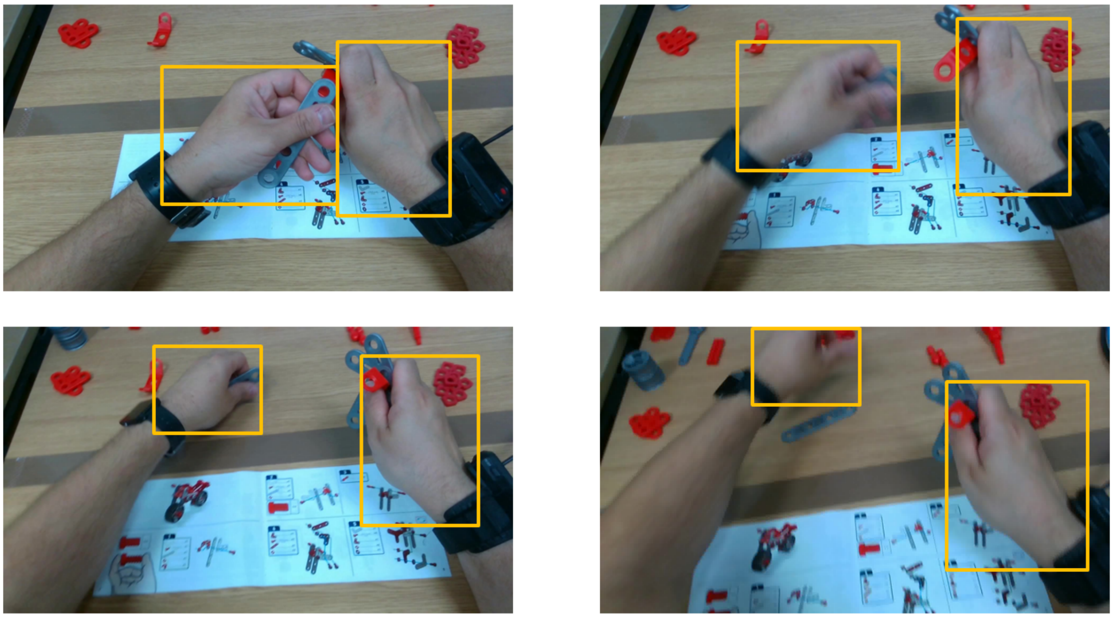


Figure 4.26: Example of hand annotations for an interaction.

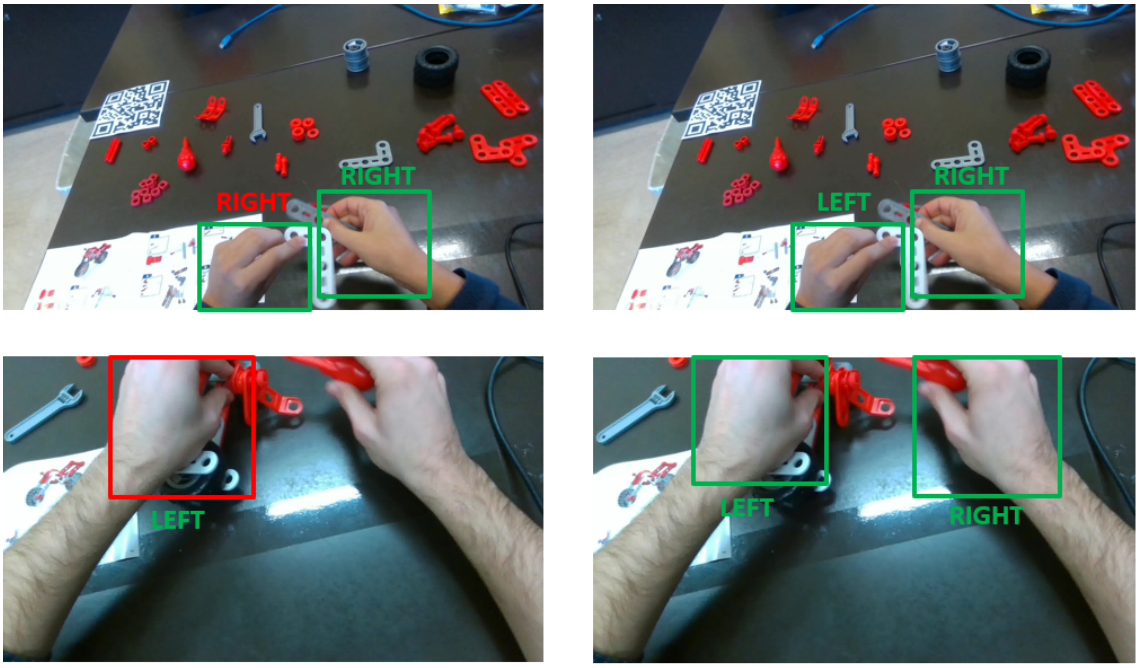
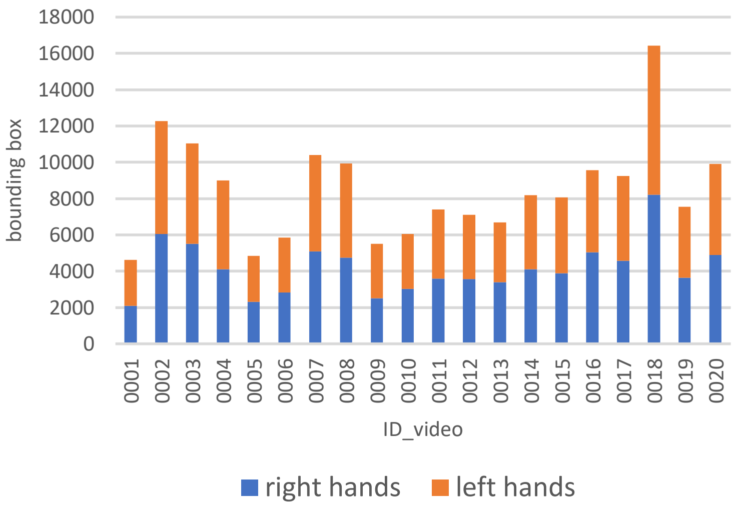
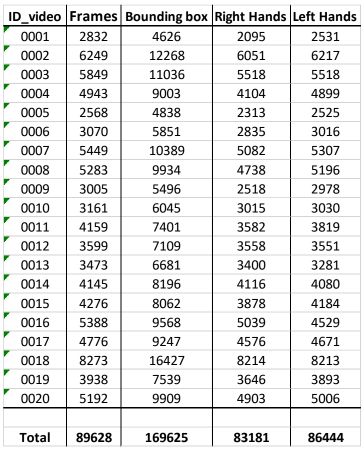


Figure 4.27: Example of the labeling procedure of the hands.





ID_video	Frames	Bounding box	Right Hands	Left Hands
0001	2832	4626	2095	2531
0002	6249	12268	6051	6217
0003	5849	11036	5518	5518
0004	4943	9003	4104	4899
0005	2568	4838	2313	2525
0006	3070	5851	2835	3016
0007	5449	10389	5082	5307
0008	5283	9934	4738	5196
0009	3005	5496	2518	2978
0010	3161	6045	3015	3030
0011	4159	7401	3582	3819
0012	3599	7109	3558	3551
0013	3473	6681	3400	3281
0014	4145	8196	4116	4080
0015	4276	8062	3878	4184
0016	5388	9568	5039	4529
0017	4776	9247	4576	4671
0018	8273	16427	8214	8213
0019	3938	7539	3646	3893
0020	5192	9909	4903	5006
Total	**89628**	**169625**	**83181**	**86444**

Figure 4.28: Hands annotations distribution.

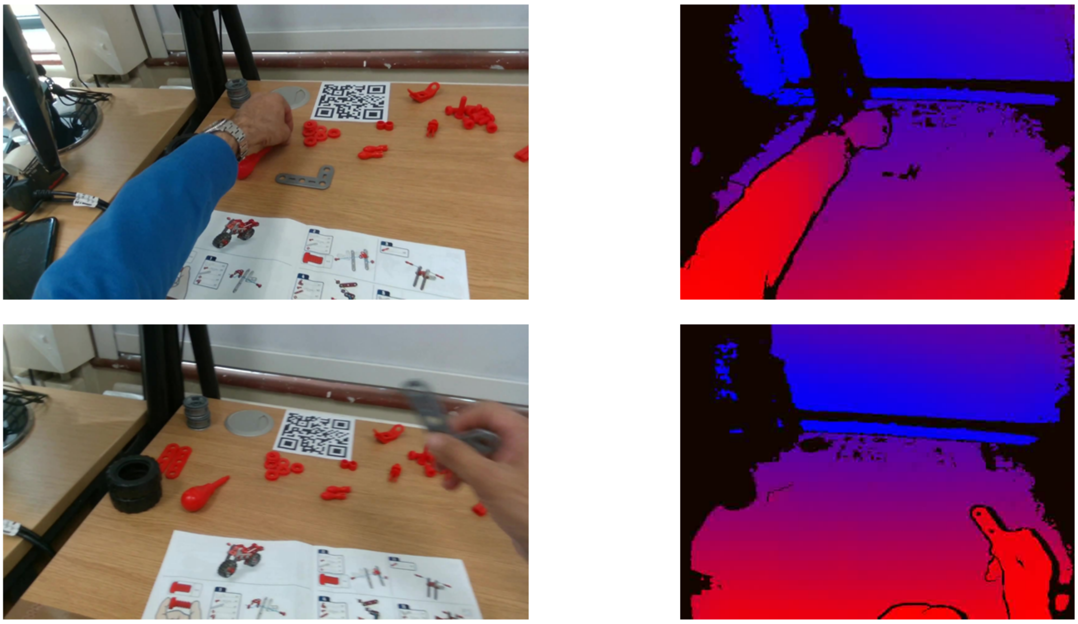


Figure 4.29: Examples of RGB and depth frames pairs.

Depth Alignment

We acquired 20 depth videos associated to the 20 RGB videos acquired with the Intel RealSense SR300. There is a constant temporal misalignment of 0.4s between depth and RGB signals due to the fact that the streams have been acquired with two different sensors (depth sensor and RGB sensor). We temporally aligned the two streams obtaining a total of 301016 depth frames. Examples of RGB frames associated with the depth maps are shown in Figure 4.29

Gaze Alignment

We acquired the gaze signal associated to the 20 RGB videos using a Pupil Core device. The gaze data has been saved with the (x, y) 2D pixel coordinates related to the RGB frame of the RealSense, as well as the related confidence score and the timestamp. For each RGB frame, we associated a gaze signal selecting only data with a confidence $\geqslant$ 0.6 and considering the timestamp related to the considered frame (see Figure 4.30).

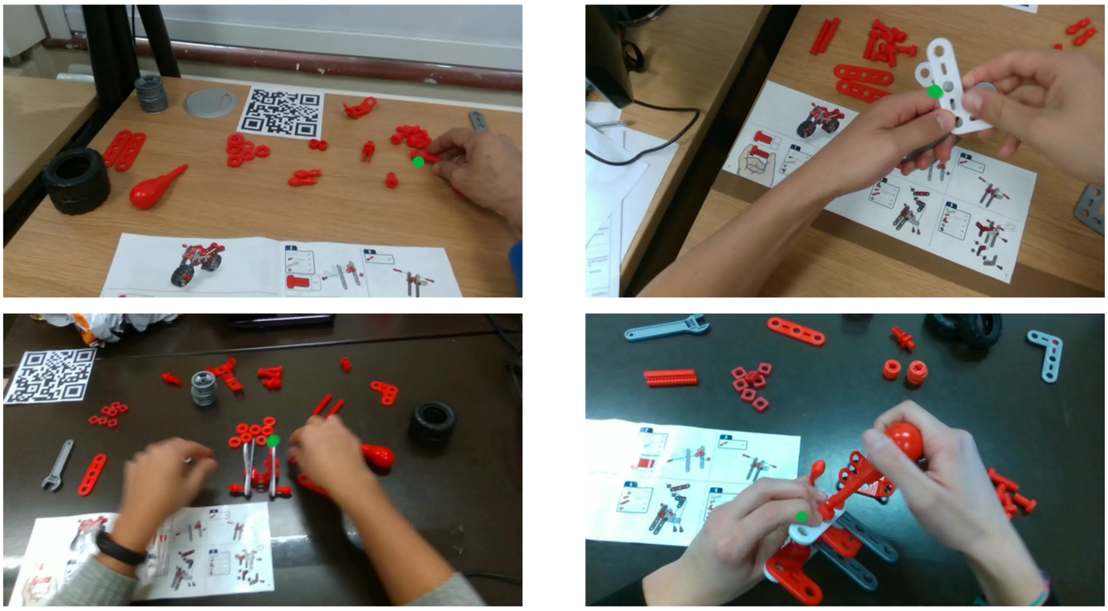


Figure 4.30: Examples of RGB frames with the associated gaze signal.

Split	#Videos	Duration (min)	%	#EHOIs Segments	Bounding Boxes	Country (U.K/Italy)	Table (Light/Dark)
Train	11	236.47	55%	5057	37386	6/5	6/5
Val	2	46.57	10%	977	6983	1/1	1/1
Test	7	134.93	35%	2824	19980	4/3	4/3

Table 4.1: Statistics of the three splits: Train, Validation and Test.

4.4 Benchmarks and Results on the MECCANO Dataset

The MECCANO dataset is suitable to study a variety of tasks, considering the challenging industrial-like scenario in which it was acquired. We considered four tasks related to human-object interaction understanding for which we provide baseline results: 1) *Action Recognition*, 2) *Active Object Detection*, 3) *Active Object Recognition* and 4) *Egocentric Human-Object Interaction (EHOI) Detection.* While some of these tasks have been considered in previous works, none of them has been studied in industrial scenarios from the egocentric perspective. Moreover, it is worth noting that the EHOI Detection task has never been treated in previous works. We split the dataset into three subsets (*Training, Validation* and *Test*) designed to balance the different types of desks (light, dark) and countries in which the videos have been acquired (IT, U.K.). Table 4.1 reports some statistics about the three splits, such as the number of videos, the total duration (in seconds), the number of temporally annotated EHOIs and the number of bounding box annotations.

4.4.1 Action Recognition

Action Recognition consists in determining the action performed by the camera wearer from an egocentric video segment. Specifically, let $C_a = \{c_1, c_2, ..., c_n\}$ be the set of action classes and let $A_i = [t_{s_i}, t_{e_i}]$ be a video segment, where t_{s_i} and t_{e_i} are the start and the end times of the action respectively. The aim is to assign the correct action class $c_i \in C_a$ to the segment A_i.

Evaluation Measures

We evaluate action recognition using Top-1 and Top-5 accuracy computed on the whole test set. As class-aware measures, we report class-mean precision, recall and F_1-score.

	Top-1 Accuracy	Top-5 Accuracy	Avg Class Precision	Avg Class Recall	Avg Class F_1-score
C2D [167]	41.92	71.95	37.6	38.76	36.49
I3D [97]	42.51	72.35	40.04	40.42	38.88
SlowFast [104]	**42.85**	**72.47**	**42.11**	**41.48**	**41.05**

Table 4.2: Baseline results for the action recognition task.

Methods and Implementation Details

We considered 2D CNNs as implemented in the PySlowFast library [167] (C2D), I3D [97] and SlowFast [104] networks, which are state-of-the-art methods for action recognition. In particular, for all baselines we used the PySlowFast implementation based on a ResNet-50 [168] backbone pre-trained on Kinetics [117]. The SlowFast, C2D and I3D baselines all require fixed-length clips at training time. Hence, we temporally downsample or upsample uniformly each video shot before passing it to the input layer of the network. The average number of frames in a video clip in the MECCANO dataset is 26.19. For SlowFast network, we set $\alpha = 4$ and $\beta = \frac{1}{8}$. We set the batch-size to 12 for C2D and I3D, we used a batch-size of 20 for SlowFast. We trained C2D, I3D and SlowFast networks on 2 NVIDIA V100 GPUs for 80, 70 and 40 epochs with learning rates of 0.01, 0.1 and 0.0001 respectively. These settings allowed all baselines to converge.

Results

Table 4.2 reports the results obtained by the baselines for the action recognition task. All baselines obtained similar performance in terms of Top-1 and Top-5 accuracy with SlowFast networks achieving slightly better performance. Interestingly, performance gaps are more consistent when we consider precision, recall and F_1 scores, which is particularly relevant given the long-tailed distribution of actions in the proposed dataset (see Figure 4.17). Note that, in our benchmark, SlowFast obtained the best results with a Top-1 accuracy of 47.82 and an F_1-score of 41.05. Figure 4.31 shows some qualitative results of the SlowFast baseline. Note that, in the second and third example, the method predicts correctly only the verb or the object. In general, the results suggest that action recognition with the MECCANO dataset is challenging and offers a new scenario to compare action recognition algorithms.

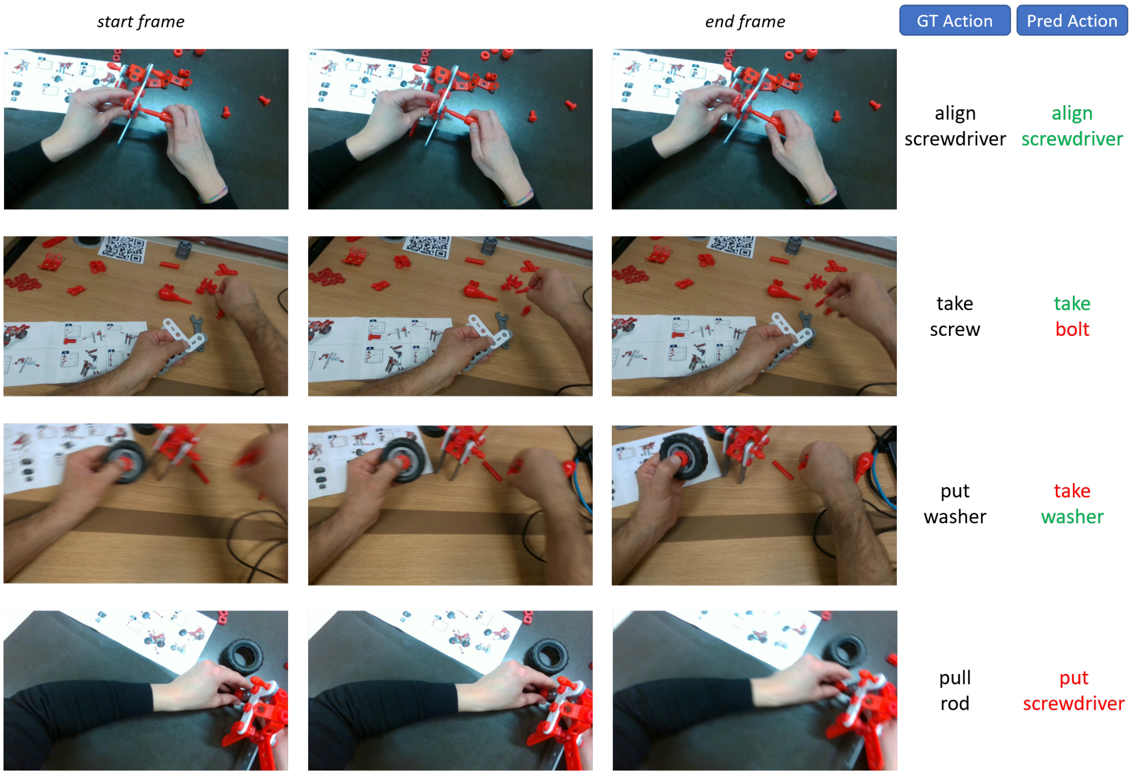


Figure 4.31: Qualitative results for the action recognition task. Correct predictions are in green while wrong predictions are in red.

4.4.2 Active Object Detection

The aim of the Active Object Detection task is to detect all the *active* objects involved in EHOIs. Let $O_{act} = \{o_1, o_2, ..., o_n\}$ be the set of *active* objects in the image. The goal is to detect with a bounding box each *active* object $o_i \in O_{act}$.

Evaluation Measures

As evaluation measure, we use Average Precision (AP) (we use the AP because we considered only the general *active object* class), which is used in standard object detection benchmarks. We set the IoU threshold equal to 0.5 in our experiments.

Methods and Implementation Details

To address the problem of recognizing active objects, the Hand-Object Detector proposed in [112] has been considered. The model has been designed to detect hands and objects when they are in contact. This architecture is based on Faster-RCNN [58] and predicts a box around the visible human hands, as well as boxes around the objects the hands are in contact with and a link between them. We used the Hand-Object Detector [112] pretrained on EPIC-Kitchens [106], EGTEA [121] and CharadesEGO [169] as provided by the authors [112]. The model has been trained to recognize hands and to detect the *active* objects regardless of their class. Hence, it should generalize to others domains. With default parameters, the Hand-Object Detector can find at most two *active* objects in contact with hands. Since our dataset tends to contain more *active* objects in a single EHOI (up to 7), we consider two variants of this model by changing the threshold on the distance between hands and detected objects. In the first variant, the threshold is set to the average distance between hands and *active* objects on the MECCANO dataset. We named this variant "*Avg distance*". In the second variant, we removed the thresholding operation and considered all detected objects as *active* objects. We named this variant "*All objects*". We further adapted the Hand-Object Detector [112] re-training the Faster-RCNN component to detect all *active* objects of the MECCANO dataset. Faster-RCNN has been trained on the training and validation sets using the provided *active* object labels. We set the learning rate to 0.005 and trained Faster-RCNN with a ResNet-101 backbone and Feature Pyramid Network for 100K iterations on 2 NVIDIA V100 GPUs. We used the Detectron2 implementation [170]. The model is trained to recognize objects along with their classes. However, for the active object detection task, we ignore output class names and only consider a single "active object" class.

Results

Table 4.3 shows the results obtained by the *active* object detection task baselines. The results highlight that the Hand-Object Detector [112] is not able to generalize to a domain different than the one on which it was trained. All the three variants of the Hand-Object Detector using the original object detector obtained an AP approximately equal to 11% (first three rows of Table 4.3). Re-training the object detector on the MECCANO dataset allowed to improve performance by significant margins. In particular, using the standard distance threshold value, we obtained an AP of 20.18%. If we consider the average distance as the threshold to discriminate *active*

Method	**AP (IoU >0.5)**
Hand Object Detector [112]	11.17%
Hand Object Detector [112] (Avg dist.)	11.10%
Hand Object Detector [112] (All dist)	11.34%
Hand Object Detector [112] + Objs re-training	20.18%
Hand Object Detector [112] + Objs re-training (Avg dist.)	33.33%
Hand Object Detector [112] + Objs re-training (All dist.)	**38.14%**

Table 4.3: Baseline results for the *active* object detection task.

and *passive* objects, we obtain an AP of 33.33%. Removing the distance threshold (last row of Table 4.3), allows to outperform all the previous results obtaining an AP equal to 38.14%. This suggests that adapting the general object detector to the challenging domain of the proposed dataset is key to performance. Indeed, training the object detector to detect only *active* objects in the scene already allows to obtain reasonable results, while there still space for improvement.

4.4.3 Active Object Recognition

The task consists in detecting and recognizing the *active* objects involved in EHOIs considering the 20 object classes of the MECCANO dataset. Formally, let $O_{act} = \{o_1, o_2, ..., o_n\}$ be the set of *active* objects in the image and let $C_o = \{c_1, c_2, ..., c_m\}$ be the set of object classes. The task consists in detecting objects $o_i \in O_{act}$ and assigning them the correct class label $c \in C_o$.

Evaluation Measures

We use mAP [171] with threshold on IoU equal to 0.5 for the evaluations.

Method and Implementation Details

As a baseline, we used a standard Faster-RCNN [58] object detector. For each image the object detector predicts *(x, y, w, h, class)* tuples which represent the object bounding boxes and the associated classes. We used the same model adopted for the Active Object Detection task, retaining also object classes at test time.

ID	Class\Video	0008	0009	0010	0011	0012	0019	0020	AP (per class)
0	instruction booklet	62.00%	38.78%	42.97%	63.75%	29.84%	38.25%	47.65%	46.18%
1	gray_angled_perforated_bar	9.55%	18.81%	14.72%	2.17%	16.42%	0%	6.89%	9.79%
2	partial_model	35.68%	31.74%	35.82%	42.55%	32.16%	33.02%	43.80%	36.40%
3	white_angled_perforated_bar	43.70%	39.86%	9.90%	45.32%	24.94%	16.35%	33.31%	30.48%
4	wrench	//	//	//	11.11%	//	10.43%	//	10.77%
5	screwdriver	61.82%	57.68%	68.57%	54.21%	57.14%	62.68%	61.37%	60.50%
6	gray_perforated_bar	19.36%	40.26%	30.89%	53.06%	29.68%	26.82%	15.76%	30.83%
7	wheels_axle	11.37%	18.34%	04.63%	1.79%	31.61%	03.91%	04.35%	10.86%
8	red_angled_perforated_bar	18.65%	01.57%	4.81%	00.09%	12.27%	05.98%	09.64%	07.57%
9	red_perforated_bar	23.35%	26.69%	34.72%	24.58%	20.70%	11.21%	17.91%	22.74%
10	rod	14.90%	07.40%	22.41%	19.73%	15.57%	17.84%	14.04%	15.98%
11	handlebar	44.39%	36.31%	28.79%	26.92%	12.50%	27.27%	52.48%	32.67%
12	screw	48.64%	42.87%	40.00%	16.96%	44.99%	43.88%	35.35%	38.96%
13	tire	45.93%	71.68%	63.09%	89.01%	37.83%	39.69%	65.15%	58.91%
14	rim	45.10%	35.71%	42.57%	59.26%	22.28%	90.00%	57.54%	50.35%
15	washer	31.52%	39.39%	19.00%	19.57%	53.43%	44.45%	09.06%	30.92%
16	red_perforated_junction_bar	19.28%	13.51%	07.55%	30.74%	28.63%	22.02%	16.89%	19.80%
17	red_4_perforated_junction_bar	24.20%	43.50%	39.11%	85.71%	44.23%	28.37%	20.62%	40.82%
18	bolt	33.14%	33.61%	11.29%	17.16%	28.46%	21.31%	19.12%	23.44%
19	roller	09.93%	40.50%	28.15%	5.76%	0.23%	18.20%	09.36%	16.02%
	mAP (per video)	31.71%	33.59%	28.89%	33.47%	28.57%	28.08%	28.44%	**30.39%**

Table 4.4: Baseline results for the *active* object recognition task. We report the AP values for each class which are the averages of the AP values for each class of the Test videos. In the last column, we report the mAP per class, which is the average mAP of the Test videos.

Results

Table 4.4 reports the results obtained with the baseline in the *Active* Object Recognition task. We report the AP values for each class considering all the videos belonging to the test set of the MECCANO dataset. The last column shows the average of the AP values for each class and the last row reports the mAP values for each test video. The mAP was computed as the average of the mAP values obtained in each test video. AP values in the last column show that large objects are easier to recognize (e.g. *instruction booklet: 46.48%; screwdriver: 60.50%; tire: 58.91%; rim: 50.35%*). Performance suggests that the proposed dataset is challenging due to the presence of small objects.

Figure 4.32 reports some qualitative results for this task. In particular, in the first row, we report the correct *active* object predictions, while in the second row we report two examples of wrong predictions. In the wrong predictions, the right *active* object is recognized but other *passive* objects are wrongly detected and recognized as *active* (e.g., instruction booklet in the example bottom-left or the red bars in the example bottom-right of Figure 4.32).

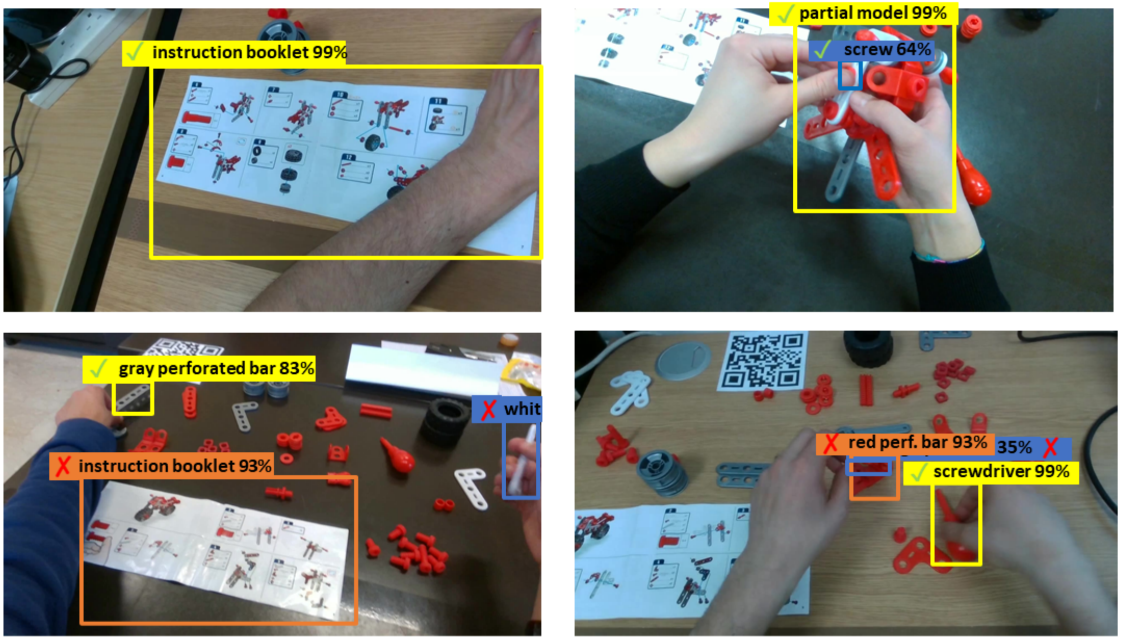


Figure 4.32: Qualitative results for the *active* object recognition task.

4.4.4 Egocentric Human-Objects Interaction (EHOI) Detection

The goal of this task is to determine egocentric human-object interactions (EHOI) in each image. Given the definition of EHOIs as <verb, objects> pairs (see Equation 4.1), methods should detect and recognize all the *active* objects in the scene, as well as the verb describing the action performed by the human.

Evaluation Measures

Following [8, 9], we use *"role AP"* as an evaluation measure. Formally, a detected EHOI is considered as a true positive if 1) the predicted object bounding box has a IoU of 0.5 or higher with respect to a ground truth annotation and 2) the predicted verb matches with the ground truth. Note that only the *active* object bounding box location (not the correct class) is considered in this measure. Moreover, we used different values of IoU (e.g., 0.5, 0.3 and 0.1) to compute the *"role AP"*.

Methods and Implementation Details

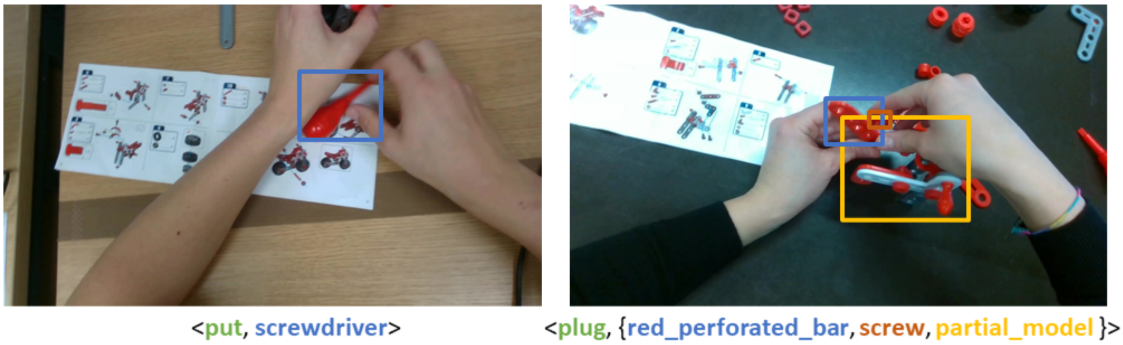
We adopted three baselines for the EHOI detection task. The first one is based on InteractNet [9], which is composed by three branches: 1) the "human-branch" to detect the humans in the scene, 2) the "object-branch" to detect the objects and 3) the "interaction-branch' which predicts the verb of the interaction focusing on the humans and objects appearance. The second one is an extension of InteractNet which also uses context features derived from the whole input frame to help the "interaction-branch" in verb prediction. The last baseline is based on the combination of a SlowFast network [167] trained to predict the verb of the EHOI considering the spatial and temporal dimensions, and Faster-RCNN [58] which detects and recognizes all *active* objects in the frame. For the "SlowFast + Faster-RCNN" baseline, we trained SlowFast network to recognize the 12 verb classes of the MECCANO dataset using the same settings as the ones considered for the action recognition task. We trained the network for 40 epochs and obtained a verb recognition Top-1 accuracy of 58.04% on the Test set. For the object detector component, we used the same model trained for the active object recognition task.

For the "human-branch" of the "InteractNet" model, we used the Hand-Object Detector [112] to detect hands in the scene. The object detector trained for active object recognition has been used for the "object-branch". The MLPs used to predict the verb class form the appearance of hands and active objects are composed by an input linear layer (e.g., 1024-d for the hands MLP and 784-d for the objects one), a ReLU activation function and an output linear layer (e.g., 12-d for both MLPs). We fused by late fusion the output probability distributions of verbs obtained from the two MLPs (hands and objects) to predict the final verb of the EHOI. We jointly trained the MLPs for $50K$ iterations on an Nvidia V100 GPU, using a batch size of 28 and a learning rate of 0.0001.

In "InteractNet + Context", we added a third MLP which predicts the verb class based on context features. The context MLP has the same architecture of the others MLPs (hands and objects) except the input linear layer which is 640-d. In this case, we jointly trained the three MLPs (hands, objects and context) for $50K$ iterations on a TitanX GPU with a batch size equal to 18 and the learning rate equal to 0.0001. The outputs of the three MLPs are hence fused by late fusion.

	mAP role		
Model	**IoU $\geq$ 0.5**	**IoU $\geq$ 0.3**	**IoU $\geq$ 0.1**
InteractNet [9]	04.92%	05.30%	05.72%
InteractNet [9] + Context	08.45%	09.01%	09.45%
SlowFast [104] + Faster-RCNN [58]	**25.93%**	**28.04%**	**29.65%**

Table 4.5: Baseline results for the EHOI detection task.


<put, screwdriver>

<plug, {red_perforated_bar, screw, partial_model }>

Figure 4.33: Qualitative results for the EHOI detection task.

Results

Table 4.5 reports the results obtained by the baselines on the test set for the EHOI detection task. The InteractNet method obtains low performance on this task with a mAP role of 4.92%. Its extension with context features, slightly improves the performance with a mAP role of 8.45%, whereas SlowFast network with Faster-RCNN achieved best results with a mAP equal to 25.93%. The results highlight that current state-of-the-art approaches developed for the analysis of still images in third person scenarios are unable to detect EHOIs in the proposed dataset, which is likely due to the presence of multiple tiny objects involved simultaneously in the EHOI and to the actions performed. On the contrary, adding the ability to process video clips with SlowFast allows for significant performance boosts. Figure 4.33 shows qualitative results obtained with the SlowFast+Faster-RCNN baseline. Note that in the second example the method correctly predicted all the objects involved simultaneously in the EHOI. Despite promising performance of the suggested baseline, the proposed EHOI detection task needs more investigation due to the challenging nature of the considered industrial-like domain.

4.5 Next-Active Objects Detection

Predicting what a user will do in the future, allows a system to support humans during their activity in any domain. Anticipating what a worker will do and which objects he will interact with in an industrial domain allows to improve safety in a factory, for example by notifying the user with an alert in case of a future dangerous action. Moreover, a system could implement an energy saving strategy automatically turning on the work tools automatically when it anticipates an interaction between the worker and the work tool.

In this chapter,, then, we define the Next-Active Objects Detection problem from the egocentric point of view (Section 4.5). In Section ?? we describe the extension of the MECCANO dataset presented in Section 4.3, called MECCANO Multimodal. The dataset is characterized by the multi-modal signals considered (i.e., RGB, depth and gaze) and by a new set of annotations useful to study the next-active objects detection problem. Finally, Section 4.4 reports the results of preliminary experiments on the MECCANO Multimodal dataset, analyzing next-active objects detection problem by considering both single frames and videos inputs.

In this section, we define the problem of Next-Active Objects Detection whose goal is to predict and localize the objects that will be involved in a future human-object interactions from the first person view. In particular, these objects which we called *next-active objects*, will be *active* when there is a physical contact. After exploring the difference between actions and interactions (Section 4.2), we defined two kinds of interactions: 1) Human - Objects Interaction (H-O) and 2) Human - Object - Object Interaction (H-O-O). The H-O interaction is represented by the contact between the hands and the target objects, whereas, the H-O-O interaction is represented by the contact between objects. Figure 4.34 shows an example of the two types of interactions.

Let $O_{act} = \{o_1, o_2, ..., o_n\}$ be the set of active objects classes and let $Rel = \{$H-O, H-O-O$\}$ be the set of the relation classes. Let T_a be the *anticipation time*, i.e. how far in advance we wish to anticipate the active objects involved in the interaction, and T_o be the *observation time*, i.e. the length of the observed video segment preceding the interaction. Given an interaction video segment $I_j = [T_{sj}, T_{ej}]$, the goal of the Next-Active Objects Detection task is to predict the relation class $r_j \in Rel$, the set of active objects involved in the interaction $I_j, O_{Ij} = \{o_i\}_{i=1}^{n}$, with $o_i \in O_{act}$,

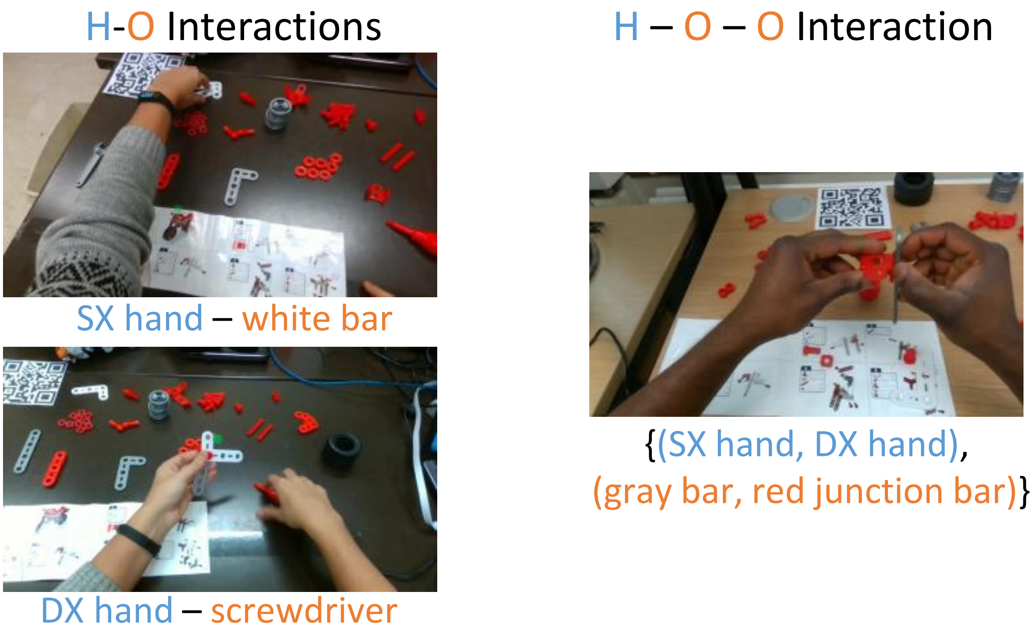


Figure 4.34: Examples of the defined interactions (H-O) and (H-O-O).

and their bounding boxes $B_{Ij} = \{b_{oi}\}_{i=1}^{n}$ where $b_{oi} = (x, y, w, h)$ is the bounding box related to the object o_i, by observing the video segment $T_o = [T_{sj} - (T_a + T_o), T_{sj} - T_a]$ (see Figure 4.35).

The task of next-active objects detection can be tackled at different levels. For example, predicting also the verb which characterizes the interaction or estimating how much time remains before the interaction starts (i.e. time to contact). In this work, we tackled the problem of predicting and localizing which objects will be active considering the MECCANO Multimodal dataset described in Section **??**.

4.5.1 Experimental Settings and Results

In this section, we report preliminary experiments related to the next-active objects detection task on the MECCANO Multimodal dataset. We firstly explored frame-based approaches analyzing the performance of the Faster-RCNN object detector on this task (Section 4.5.1). Then, we designed an architecture based on video input to include the temporal information for the next-active object detection task (Section 4.5.2).

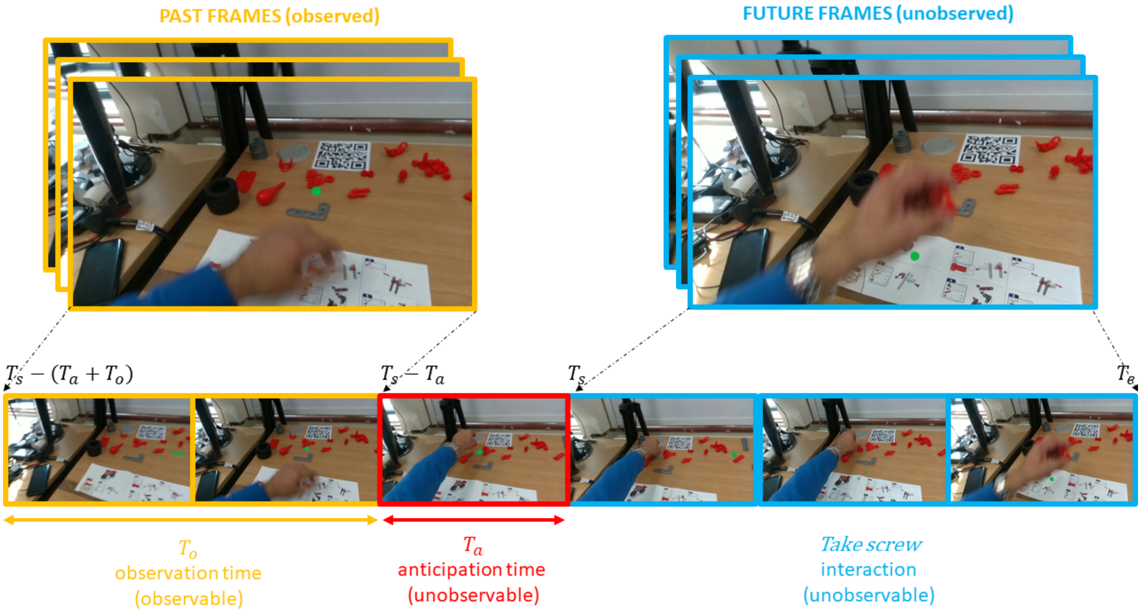


Figure 4.35: Next Active Objects Task.

Frame-based Detection

We tackled the Next-Active Objects Detection task considering the following five variants of the standard Faster-RCNN [58] object detector. We trained all the object detectors using the Detectron2 framework [170].

Faster-RCNN (active objects) This object detector has been trained only with the active objects involved in the interaction when it is happening. This approach is the same used to perform the *active object recognition* task in Section 4.4.3.

Faster-RCNN (active objects) + finetuning (next active objects) This approach is based on the previous detector but we fine-tuned the object detector using the annotations of the next active objects described in Section 4.3.2. We performed the fine-tuning using the past frames of the 3848 interactions belonging to the Training set. We trained the model with the *next active objects* annotations on 4 NVIDIA V100 GPUs for 50000 iterations using a learning rate of 0.005 and a batch-size of 28.

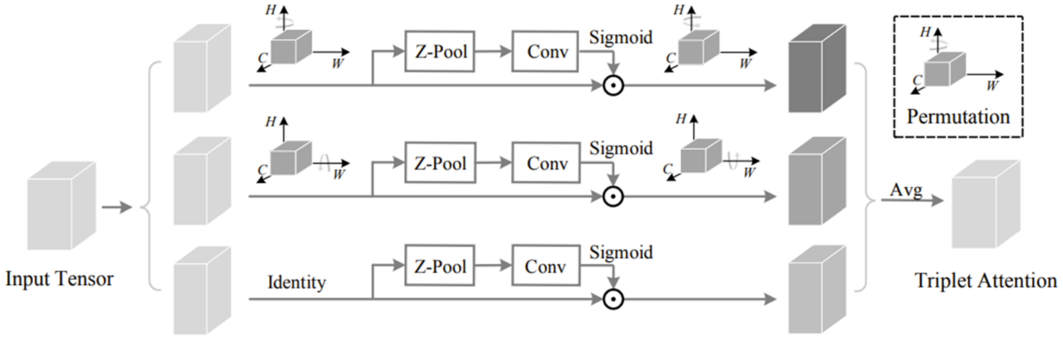


Figure 4.36: The architecture of the Faster-RCNN with triplet attention approach.

Faster-RCNN (next active objects) The object detector has been trained using only the next active objects annotations belonging to the Training set. This model has been trained with the same hyper-parameters of the previous but for 100000 iterations.

Faster-RCNN (active objects + next active objects) This approach is based on the Faster-RCNN object detector and it has been trained using both active and next active objects annotations in the training phase. Also this object detector has been trained for 100000 iterations with the same hyper-parameters of the previous.

Faster-RCNN-Triplet_Attention (next active objects) The last frame-based approach is based on Faster-RCNN and includes the additional attention mechanism proposed in [172]. This method is composed of three branches: the first branch is responsible for computing attention weights along the channel dimension C and the spatial dimension W. In the same way, the second branch is responsible for channel dimension C and spatial dimension H. A rotation operation is introduced to build connections between the channel dimension and the spatial dimensions in these two branches. The third branch is used to capture spatial dependencies (H and W). The architecture is shown in Figure 4.36. This model has been trained on 1 NVIDIA V100 gpu with a learning rate of 0.02 and a batch size of 16.

Table 4.6: Comparison of the 5 considered approaches based on the Faster-RCNN object detector.

Method ID	Method	mAP	mAP@50	mAP@75	mAPs	mAPm	mAPl
0	Faster-RCNN (active objects)	**14.10**	**26.00**	**13.30**	01.90	**07.80**	14.50
1	Faster-RCNN (active objects) + finetuning (next active objects)	11.6	19.90	12.00	**02.60**	06.00	13.10
2	Faster-RCNN (next active objects)	09.90	18.20	09.50	01.40	06.40	11.00
3	Faster-RCNN (active objects + next active objects)	14.08	25.79	12.98	02.10	07.60	**15.00**
4	Faster-RCNN-Triplet_Attention (next active objects)	11.80	22.50	10.90	01.90	06.10	13.10

Object Detectors Comparison

We tested the five approaches previously discussed on the Test set, evaluating the performance using mean Average Precision (mAP) [171], [31]. In particular we used the COCO mAP [31] which is calculated over 10 Intersection over Union (IoU) levels (i.e. from 0.5 to 0.95 with a step size of 0.05) and the standard mAP [171] with an IoU of 0.5 (mAP@50) and 0.75 (map@75). Moreover, we report the mAP calculated across different scales [31]: mAP small (mAPs) is calculated considering the objects with an area $< 32^2$ px (small objects), mAP medium (mAPm) considers only objects with $32^2 <$ area $< 96^2$ px (medium objects) and mAP large (mAPl) is calculated for large objects with an area $> 96^2$ px. Table 4.6 reports the comparison between the five considered approaches. In general, the results are very low for all the considered approaches due to the fact that this task is challenging. The method trained only with active objects (1st row) obtains the best performance considering the mAP (14.10) and the mAP@50 (26.00) measures (2nd and 3rd column). The object detector trained with both *active* and *next active objects* (3rd row) obtained similar performance with all measures. The mAP calculated across different scales are reported in the 5th, 6th and 7h columns considering small, medium and large objects respectively. Faster-RCNN pretrained on *active objects* and finetuned with *next active objects* (2nd row) obtained the best performance for the detection of small objects (5th column) with a mAPs of 2.6. Note that the MECCANO dataset is composed of many small objects (e.g. screws, bolts, etc.). The triplet attention mechanism did not improve the performance of the standard object detector (5th row). In Figure 4.37, we show some qualitative results of the best approach (considering the mAP) based on the Faster-RCNN object detector.

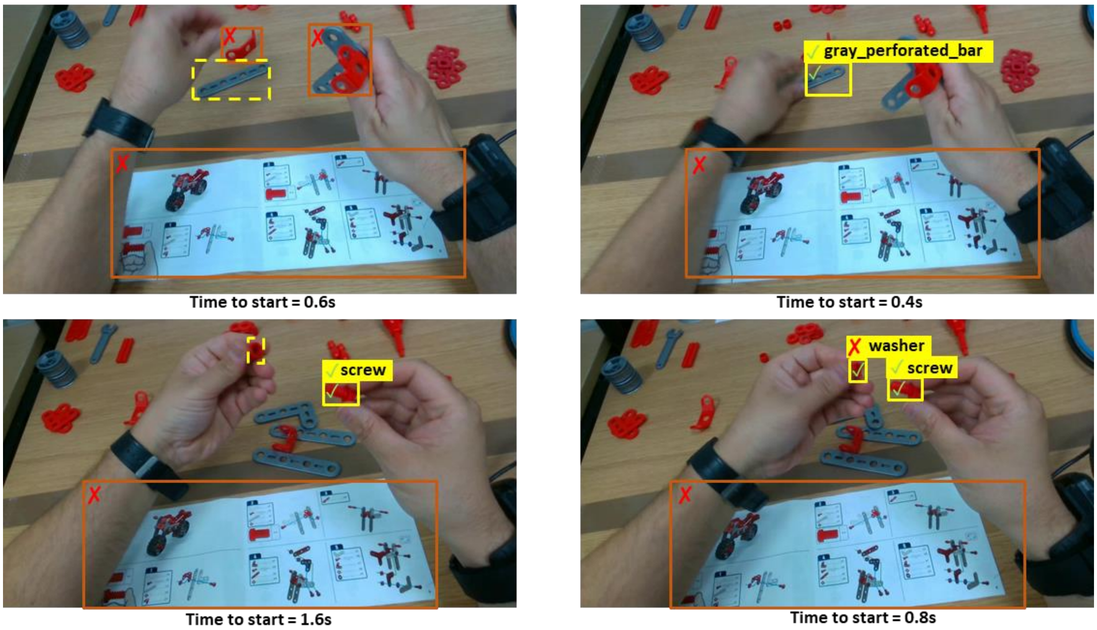


Figure 4.37: Qualitative results of the best approach based on Faster-RCNN. The dashed bounding box indicates a ground truth object which has not been detected.

4.5.2 Video-based Detection

To model the temporal relations between the frames and extract some representative feature map which encode this relation over the time dimension, we investigated some approaches considering as input video-shots related to the past of an interaction. We adapted the action detection task described in [104], where the action is predicted and also localized with a bounding box, to the *next active objects* detection task where the objects which will be active should be predicted and localized. For each frame annotated with the *next active objects* bounding boxes, we defined a video-clip. Starting from the chosen frame we go backward by 31 frames, obtaining a video-shot composed by 32 frames (which corresponds to 2.67 seconds).

We considered the SlowFast [104] architecture based on 3D CNNs to take into account video-shots. In particular, we considered SlowFast to perform a detection task using the predictions computed by an object detector on the last frame of the video clip. As object detector we choose the best object detector respect the considered five variants described in Section 4.5.1.

For each video clip related to the past of an interaction, the 3D CNN predicts if an object will be active or not with the help of the bounding box predictions related to the objects present in the last frame of the shot. We expect that these predictions could help the objects with low scores coming from the object detector, to have a chance to move up in the final ranking score. Indeed, we weigh the scores coming from both the object detector and the SlowFast network to obtain a new ranking of scores (Section 4.5.2).

Figure 4.38 shows the adopted approach to predict the *next active objects*.

Next-active vs. Not Next-active objects

In this section we analyze the performance of the SlowFast network considering the binary classification task, in which we want to predict for each detected bounding box if it represents a *next active object* or not. We trained two different SlowFast networks which differ for the input modalities used for the training phase. The first variant has been trained using the video-clips composed of the RGB frames. The second one has been trained using also the gaze signal. The gaze signal has been included in the video clips plotting for each frame a circle representing the point of the scene where the participant was watching (see Figure 4.39).

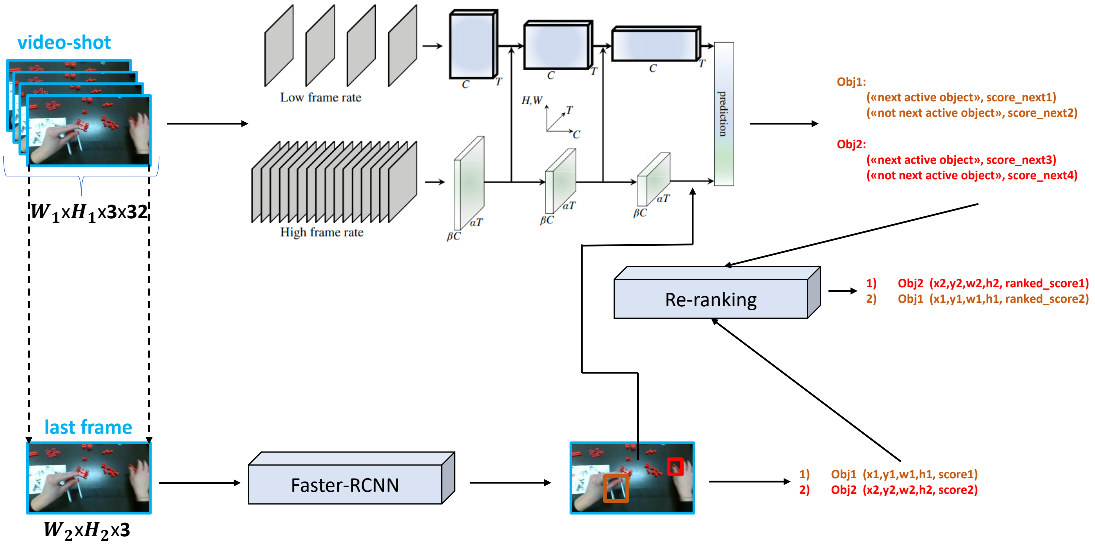

video-shot
W_1xH_1x3x32
Low frame rate
High frame rate
C
T
H,W
T
C
αT
βC
prediction
Obj1:
(«next active object», score_next1)
(«not next active object», score_next2)
Obj2:
(«next active object», score_next3)
(«not next active object», score_next4)
Re-ranking
1) Obj2 (x2,y2,w2,h2, ranked_score1)
2) Obj1 (x1,y1,w1,h1, ranked_score2)
last frame
W_2xH_2x3
Faster-RCNN
1) Obj1 (x1,y1,w1,h1, score1)
2) Obj2 (x2,y2,w2,h2, score2)

Figure 4.38: Overview of the considered architecture.

The first variant, which we named SlowFast + Detection, has been trained for 7 epochs with a learning rate of 0.01 on 2 NVIDIA V100 gpus. The second one, which we named SlowFast + Detection + Gaze, has been trained for 9 epochs with a learning rate of 0.1 using 3 NVIDIA V100 gpus. For both models we used the PySlowFast [167] framework to perform the training and the test phases. To balance the training samples belonging to the two classes (*next active object* and *not next active object*) we augmented the training predictions coming from the object detector with the ground truth annotations. In this way, we augmented the number of examples belonging to the *next active object* class.

We performed the experiments on the MECCANO Multimodal Dataset. We considered the bounding boxes predicted by the object detector. For each bounding box, we check if it represents a next-active object considering the ground truth labels, otherwise, we assign to it the "not next-active object" label. This new set of labels represent the new ground truth used to evaluate the performance of the two SlowFast variants. Note that, the object detector predictions have not been filtered and indeed, they comprise predictions with low scores. We evaluated the models using the mean Average Precision (mAP) measure with a IoU threshold of 0.5. Table 4.7 reports the performance of the two variants of SlowFast considering

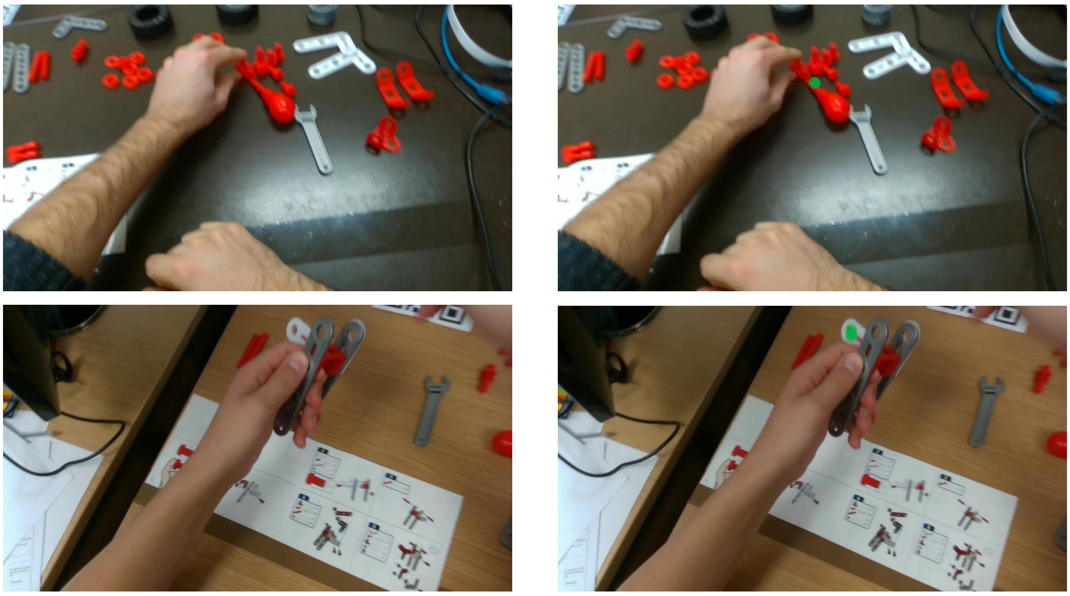


Figure 4.39: Examples of frames without the gaze plotted (left) and the correspondent version with the gaze plotted (right).

Table 4.7: Experimental results on the binary classification considering the SlowFast 3D CNN.

Method	mAP	AP (next active object)	AP (not next active object)	TestSet
SlowFast + Detection	18.90	**31.51**	6.29	without gaze
SlowFast + Detection + Gaze	**19.10**	31.10	**7.12**	without gaze
SlowFast + Detection + Gaze	19.01	31.28	6.75	with gaze

the mAP and also the AP per class. The performance is similar. The difference between the two variants, considering the TestSet without the plotted gaze, is equal to 0.2 (mAP measure).

We also perfomed experiments considering as ground truth the filtered predictions with a score-based threshold $\geqslant 0.5$. Table 4.8 reports the performance of the two approaches considering the filtered ground truth.

Table 4.8: Experimental results on the binary classification considering the SlowFast 3D CNN on the filtered ground truth.

Method	mAP	AP (next active object)	AP (not next active object)	TestSet
SlowFast + Detection	23.03	37.88	8.19	without gaze
SlowFast + Detection + Gaze	**23.77**	38.11	**9.43**	without gaze
SlowFast + Detection + Gaze	23.66	**38.36**	8.97	with gaze

Table 4.9: Results obtained performing the re-ranking stage at different values of the λ-terms.

Method	λ_{SF}	λ_{FRCNN}	**mAP@50**
SlowFast + Detection	1	1	23.80
	1	0	11.70
	0.30	0.70	23.80
	0.50	0.50	23.80
SlowFast + Detection + Gaze	1	1	23.40
	1	0	12.00
	0.30	0.70	23.40
	0.50	0.50	23.40

Re-ranking

In this section, we explain how the re-ranking phase has been performed to obtain the new list of *next active objects* predictions. We defined the re-ranked score as follow:

$$r\_score = (\lambda_{SF} * score_{SF}) + (\lambda_{FRCNN} * score_{FRCNN}) \qquad (4.2)$$

where $score_{SF}$ and $score_{FRCNN}$ are the predictions scores coming from the SlowFast network and the Faster-RCNN respectively, while the λ-terms are used to weight each score. In our experiments, we assigned different values for the λ-terms to weigh in different ways the two scores. Table 4.9 shows the results of the proposed approach to detect *next active objects* from videos at different values of the λ-terms. The best results are obtained for both approaches, if the score coming from the Faster-RCNN is considered. If we remove the Faster-RCNN score (2nd row) the performance significantly drop.

Table 4.10 reports the results obtained with the best frame-based and video-based approaches. The best performance are obtained with the Faster-RCNN object detector trained on the active objects (mAP@50 of 26.00).

Table 4.10: Comparison of the results obtained considering the best frame-based and video-based approaches.

Method	**mAP@50**
Faster-RCNN (active objects)	26.00
SlowFast + Detection	23.80

Chapter 5

Conclusions

In this thesis, the human behavior has been studied considering the First Person Vision (FPV) paradigm. In contrast to the Third Person Vision (TPV), data acquired from the first person point of view contains useful information to understand human behavior and assist the human in many domains. We have investigated two challenging domains (i.e., Cultural Heritage and Industrial) in which a first person vision system could assist humans providing useful services to improve their experience.

Chapter 3 analyzed the behavior of visitors in the cultural heritage domain from the first person point of view. Since there are not datasets composed of egocentric videos related to the cultural sites in the literature, we acquired and publicly released two challenging datasets which consider two real cultural sites: UNICT-VEDI and Egocentric Cultural Heritage (EGO-CH). We addressed many fundamental tasks related to the visitors behavior understanding on the proposed datasets, with the aim to build an intelligent system able to assist both visitors and cultural managers. The considered tasks are: room-based localization, points of interest recognition, semantic object segmentation, object retrieval, survey generation. We developed the VEDI system, which is a first person video systems. This system allows to improve the fruition of visitors in cultural sites providing many services which are the results of the studies reported in this thesis, considering the aforementioned problems.

Findings: Our study pointed out the following:

- The two challenging datasets UNICT-VEDI and EGO-CH are suitable to study the human behavior on cultural sites. The presented data include high variability in terms of environments, objects and different behaviors considering that they have been acquired by real visitors. We believe that UNICT-VEDI and EGO-CH can be valuable benchmarks to tackle tasks related to the cultural heritage domain;

- The room-based localization problem has been investigated reporting baseline results using a state-of-the-art method on data acquired using a head-mounted HoloLens and a chest-mounted GoPro device. Despite the larger field of view of the GoPro device, HoloLens allows to achieve similar performance in the localization task;

- We defined the concept of "Point of Interest" in the domain of cultural sites. We observed that a point of interest can be either an environment or an object. We addressed the point of interest recognition task considering over 200 different points of interest belonging to two different cultural sites. Experiments show that the adopted methods achieve complementary performance on this challenging task;

- Considering the Semantic Object Segmentation problem, we showed that the use of synthetic images can be beneficial to improve performance on real data, especially when coupled with image-to-image translation techniques, to reduce the domain shift arising from the two different data sources. The proposed dataset can also be used to study the problem of unsupervised domain adaptation for semantic object segmentation, which assumes the unavailability of real training data;

- The study on the considered fundamental tasks in cultural sites, led to build a real first person vision system (VEDI) able to assist both visitors and cultural managers in the domain of cultural heritage. This demonstrates that a wearable device allows to capture useful information with respect to a third person system, which allows to provide services to improve the humans lives.

Limitations: In the following, we report the limitations of the study described in this thesis:

- The localization task could be studied considering a point-wise information. In this way, a navigation system could assist the visitor indicating where are placed some points of interest with respect to his position or he can reach a specific area of the building with detailed indications;

- The semantic object segmentation problem in cultural sites has been studied in a supervised fashion. This assumes the need to annotate real data to solve the considered task. Unsupervised domain adaptation approaches should be taken into account with the proposed dataset;

- The developed VEDI system could be improved with the study of new tasks which allows the development of new services. The system has been developed considering data where there are not other visitors simultaneously in the cultural sites. A future direction could explore what could happen in a real case where other persons visits the cultural sites.

Chapter 4 explored the human behavior in the industrial domain. We acquired and annotated the MECCANO dataset and its second version MECCANO Multimodal, which comprises multiple input signals, to encourage the research community to study human behaviors in the challenging industrial domain. We defined two new tasks (i.e., egocentric human-object interaction and next-active objects) which are fundamental to provide assistance and guarantee the safety of the workers in this domain and we also report a benchmark on this challenging dataset addressing the following problems: action recognition, active object detection, active object recognition, egocentric human-object interaction, next-active objects prediction.

Findings: The main findings of this investigation are as follows:

- The provided definition of the Egocentric Human-Object Interaction (EHOI) is a starting point to encourage other researchers to study the interactions from the first person point of view;

- The analysis of the meaning and differences about the *action* and *interaction* concepts highlights how they are correlated but don't represent the same thing;

- The MECCANO and its extension MECCANO Multimodal datasets, which comprises multimodal signals, have been publicly released to study the human behaviors in the industrial domain. These are the first datasets acquired in this challenging domain;

- The definition of the new task of next-active object prediction from the egocentric point of view, which is suitable to improve workers safety.

Limitations: The limitations discovered in this study are:

- The hands of the humans should be included to solve the EHOI task due to the useful information which represent. The contact between the hands and the objects represent the start of the human-object interactions. Future work will explore approaches that will consider the hands for improving performance on this task;
- The next-active object task could be explored taking into account the objects and hands trajectories which could represent a strong signal useful to anticipate what will happen. Moreover, the gaze signal needs to be explored and modeled with the aim to improve performance on this task. Also, the depth maps should be used for the design of a new algorithm which predicts the next-active objects.

Bibliography

[1] A. Dey, G. Abowd, and D. Salber. “A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications”. In: *Human–Computer Interaction* 16 (2001), pp. 166 –97.

[2] S. Greenberg. “Context as a Dynamic Construct”. In: *Human–Computer Interaction* 16 (2001), pp. 257 –268.

[3] T. Kanade and M. Hebert. “First-Person Vision”. In: *Proceedings of the IEEE* 100 (2012), pp. 2442–2453.

[4] S. Mann. “Wearable Computing: A First Step Toward Personal Imaging”. In: *Computer* 30 (1997), pp. 25–32.

[5] S. Mann. “Humanistic computing: ”WearComp” as a new framework and application for intelligent signal processing”. In: 1998.

[6] A. Furnari, S. Battiato, and G. M. Farinella. “Personal-Location-Based Temporal Segmentation of Egocentric Video for Lifelogging Applications”. In: *Journal of Visual Communication and Image Representation* (2018). ISSN: 1047-3203. DOI: https://doi.org/10.1016/j.jvcir.2018.01.019.

[7] E. Spera, A. Furnari, S. Battiato, and G. M. Farinella. “EgoCart: a Benchmark Dataset for Large-Scale Indoor Image-Based Localization in Retail Stores”. In: *IEEE Transactions on Circuits and Systems for Video Technology* 31 (4 2021), pp. 1253–1267. URL: home/_paper/spera2021egocart.pdf.

[8] S. Gupta and J. Malik. “Visual Semantic Role Labeling”. In: *ArXiv* abs/1505.04474 (2015).

[9] G. Gkioxari, R. B. Girshick, P. Dollár, and K. He. “Detecting and Recognizing Human-Object Interactions”. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018), pp. 8359–8367.

[10] S. Qi, W. Wang, B. Jia, J. Shen, and S.-C. Zhu. “Learning Human-Object Interactions by Graph Parsing Neural NetworksChao2018LearningTD”. In: *ArXiv* abs/1808.07962 (2018).

[11] Y.-W. Chao, Y. Liu, X. Liu, H. Zeng, and J. Deng. “Learning to Detect Human-Object Interactions”. In: *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)* (2018), pp. 381–389.

[12] P. Zhou and M. Chi. “Relation Parsing Neural Network for Human-Object Interaction Detection”. In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 843–851.

[13] Y. Liao, S. Liu, F. Wang, Y. Chen, C. Qian, and J. Feng. “PPDM: Parallel Point Detection and Matching for Real-time Human-Object Interaction Detection”. In: *CVPR*. 2020.

[14] T. Wang, T. Yang, M. Danelljan, F. S. Khan, X. Zhang, and J. Sun. “Learning Human-Object Interaction Detection Using Interaction Points”. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020.

[15] A. Gupta, A. Kembhavi, and L. S. Davis. “Observing Human-Object Interactions: Using Spatial and Functional Compatibility for Recognition”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31.10 (2009), pp. 1775–1789.

[16] B. Yao and L. Fei-Fei. “Recognizing Human-Object Interactions in Still Images by Modeling the Mutual Context of Objects and Human Poses”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.9 (2012), pp. 1691–1703.

[17] R. Cucchiara and A. Del Bimbo. “Visions for augmented cultural heritage experience”. In: *IEEE MultiMedia* 21.1 (2014), pp. 74–82.

[18] F. Colace, M. De Santo, L. Greco, S. Lemma, M. Lombardi, V. Moscato, and A. Picariello. “A Context-Aware Framework for Cultural Heritage Applications”. In: *Signal-Image Technology and Internet-Based Systems (SITIS), 2014 Tenth International Conference on*. IEEE. 2014, pp. 469–476.

[19] G. Taverriti, S. Lombini, L. Seidenari, M. Bertini, and A. Del Bimbo. "Realtime Wearable Computer Vision System for Improved Museum Experience". In: *ACM Multimedia*. 2016, pp. 703–704.

[20] M. Portaz, M. Kohl, G. Quénot, and J.-P. Chevallet. "Fully Convolutional Network and Region Proposal for Instance Identification with egocentric vision". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 2383–2391.

[21] G Gallo, G Signorello, G. Farinella, and A Torrisi. "Exploiting Social Images to Understand Tourist Behaviour". In: *ICIAP*. 2017, pp. 707–717.

[22] L. Seidenari, C. Baecchi, T. Uricchio, A. Ferracani, M. Bertini, and A. D. Bimbo. "Deep Artwork Detection and Retrieval for Automatic Context-Aware Audio Guides". In: *ACM Transactions on Multimedia Computing, Communications, and Applications* 13.3s (2017), p. 35.

[23] G. Signorello, G. M. Farinella, G. Gallo, L. Santo, A. Lopes, and E. Scuderi. "Exploring Protected Nature Through Multimodal Navigation of Multimedia Contents". In: *ACIVS*. 2015, pp. 841–852.

[24] A. S. Razavian, O. Aghazadeh, J. Sullivan, and S. Carlsson. "Estimating Attention in Exhibitions Using Wearable Cameras". In: *2014 22nd International Conference on Pattern Recognition* (2014), pp. 2691–2696.

[25] D. Raptis, N. K. Tselios, and N. M. Avouris. "Context-based design of mobile applications for museums: a survey of existing practices". In: *Mobile HCI*. 2005.

[26] M. Portaz, J. Poignant, M. Budnik, P. Mulhem, J.-P. Chevallet, and L. Goeuriot. "Construction et évaluation d'un corpus pour la recherche d'instances d'images muséales." In: *CORIA*. 2017, pp. 17–34.

[27] F. Bartoli, G. Lisanti, L. Seidenari, S. Karaman, and A. Del Bimbo. "Museumvisitors: a dataset for pedestrian and group detection, gaze estimation and behavior understanding". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2015, pp. 19–27.

[28] P. Koniusz, Y. Tas, H. Zhang, M. T. Harandi, F. Porikli, and R. Zhang. "Museum Exhibit Identification Challenge for Domain Adaptation and Beyond". In: *CoRR* abs/1802.01093 (2018). arXiv: 1802.01093. URL: http://arxiv.org/abs/1802.01093.

[29] R. D. Chiaro, A Bagdanov, and A. D. Bimbo. "{NoisyArt}: A Dataset for Webly-supervised Artwork Recognition". In: *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications.* 2019.

[30] M. Everingham, L. Gool, C. K. Williams, J. Winn, and A. Zisserman. "The Pascal Visual Object Classes (VOC) Challenge". In: *Int. J. Comput. Vision* 88.2 (June 2010), 303–338. ISSN: 0920-5691. DOI: 10.1007/s11263-009-0275-4.

[31] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár. *Microsoft COCO: Common Objects in Context.* 2014. URL: http://arxiv.org/abs/1405.0312.

[32] S. R. Richter, V. Vineet, S. Roth, and V. Koltun. "Playing for Data: Ground Truth from Computer Games". In: *ArXiv* abs/1608.02192 (2016).

[33] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez. "The SYNTHIA Dataset: A Large Collection of Synthetic Images for Semantic Segmentation of Urban Scenes". In: *IEEE CVPR.* 2016.

[34] H. Abu Alhaija, S. K. Mustikovela, L. Mescheder, A. Geiger, and C. Rother. "Augmented Reality Meets Computer Vision: Efficient Data Generation for Urban Driving Scenes". In: *Int. J. Comput. Vision* 126.9 (Sept. 2018), 961–972. ISSN: 0920-5691. DOI: 10.1007/s11263-018-1070-x.

[35] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig. "Virtual Worlds as Proxy for Multi-object Tracking Analysis". In: *IEEE Conference on Computer Vision and Pattern Recognition* (2016).

[36] K. Curran, E. Furey, T. Lunney, J. Santos, D. Woods, and A. McCaughey. "An evaluation of indoor location determination technologies". In: *Journal of Location Based Services* 5.2 (2011), pp. 61–78.

[37] R. Want and A. Hopper. “Active badges and personal interactive computing objects”. In: *IEEE Transactions on Consumer Electronics* 38.1 (1992), pp. 10–20.

[38] Y. Gu, A. Lo, and I. Niemegeers. “A survey of indoor positioning systems for wireless personal networks”. In: *IEEE Communications surveys & tutorials* 11.1 (2009), pp. 13–32.

[39] G. Amato, F. Falchi, and C. Gennaro. “Fast image classification for monument recognition”. In: *Journal on Computing and Cultural Heritage (JOCCH)* 8.4 (2015), p. 18.

[40] T. Weyand and B. Leibe. “Visual landmark recognition from Internet photo collections: A large-scale evaluation”. In: *Computer Vision and Image Understanding* 135 (2015), pp. 1 –15. ISSN: 1077-3142. DOI: https://doi.org/10.1016/j.cviu.2015.02.002.

[41] Q. Li, J. Zhu, T. Liu, J. Garibaldi, Q. Li, and G. Qiu. “Visual landmark sequence-based indoor localization”. In: *Proceedings of the 1st Workshop on Artificial Intelligence and Deep Learning for Geographic Knowledge Discovery*. 2017, pp. 14–23.

[42] A. Oliva and A. Torralba. “Modeling the shape of the scene: A holistic representation of the spatial envelope”. In: *International journal of computer vision* 42.3 (2001), pp. 145–175.

[43] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. “Learning deep features for scene recognition using places database”. In: *Advances in Neural Information Processing Systems*. 2014, pp. 487–495.

[44] T. Starner, B. Schiele, and A. Pentland. “Visual contextual awareness in wearable computing”. In: *International Symposium on Wearable Computing*. 1998, pp. 50–57.

[45] H. Aoki, B. Schiele, and A. Pentland. “Recognizing personal location from video”. In: *Workshop on Perceptual User Interfaces*. ACM. 1998, pp. 79–82.

[46] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin. "Context-based vision system for place and object recognition". In: *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on.* IEEE. 2003, pp. 273–280.

[47] J. Shotton, B. Glocker, C. Zach, S. Izadi, A. Criminisi, and A. Fitzgibbon. "Scene coordinate regression forests for camera relocalization in RGB-D images". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 2013, pp. 2930–2937.

[48] A. Kendall, M. Grimes, and R. Cipolla. "Posenet: A convolutional network for real-time 6-dof camera relocalization". In: *Proceedings of the IEEE international conference on computer vision.* 2015, pp. 2938–2946.

[49] T. Sattler, B. Leibe, and L. Kobbelt. "Efficient & effective prioritized matching for large-scale image-based localization". In: *IEEE transactions on pattern analysis and machine intelligence* 39.9 (2017), pp. 1744–1756.

[50] T. Ishihara, K. M. Kitani, C. Asakawa, and M. Hirose. "Inference Machines for supervised Bluetooth localization". In: *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on.* 2017, pp. 5950–5954.

[51] T. Ishihara, J. Vongkulbhisal, K. M. Kitani, and C. Asakawa. "Beacon-Guided Structure from Motion for Smartphone-Based Navigation". In: *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on.* 2017, pp. 769–777.

[52] Q. Xu, L. Li, J. H. Lim, C. Y. C. Tan, M. Mukawa, and G. Wang. "A wearable virtual guide for context-aware cognitive indoor navigation". In: *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services.* ACM. 2014, pp. 111–120.

[53] R. Girshick, J. Donahue, T. Darrell, and J. Malik. "Rich feature hierarchies for accurate object detection and semantic segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition.* 2014, pp. 580–587.

[54] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. Lecun. “Overfeat: Integrated recognition, localization and detection using convolutional networks”. English (US). In: *International Conference on Learning Representations (ICLR2014), CBLS, April 2014*. 2014.

[55] R. Girshick. “Fast r-cnn”. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1440–1448.

[56] K. He, X. Zhang, S. Ren, and J. Sun. “Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition”. In: *CoRR* abs/1406.4729 (2014). arXiv: 1406.4729. URL: http://arxiv.org/abs/1406.4729.

[57] C. Szegedy, S. E. Reed, D. Erhan, and D. Anguelov. “Scalable, High-Quality Object Detection”. In: *CoRR* abs/1412.1441 (2014).

[58] S. Ren, K. He, R. Girshick, and J. Sun. “Faster R-CNN: Towards real-time object detection with region proposal networks”. In: *Advances in neural information processing systems*. 2015, pp. 91–99.

[59] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. “Ssd: Single shot multibox detector”. In: *European conference on computer vision*. Springer. 2016, pp. 21–37.

[60] K. He, G. Gkioxari, P. Dollár, and R. Girshick. “Mask r-cnn”. In: *arXiv preprint arXiv:1703.06870* (2017).

[61] J. Redmon and A. Farhadi. “YOLOv3: An Incremental Improvement”. In: *CoRR* abs/1804.02767 (2018). arXiv: 1804.02767.

[62] J. Redmon and A. Farhadi. “YOLO9000: Better, Faster, Stronger”. In: *arXiv preprint arXiv:1612.08242* (2016).

[63] Y. Wu and K. He. “Group Normalization”. In: *The European Conference on Computer Vision (ECCV)*. 2018.

[64] H. Law and J. Deng. “CornerNet: Detecting Objects as Paired Keypoints”. In: *The European Conference on Computer Vision (ECCV)*. 2018.

[65] B. Jiang, R. Luo, J. Mao, T. Xiao, and Y. Jiang. “Acquisition of Localization Confidence for Accurate Object Detection”. In: *The European Conference on Computer Vision (ECCV)*. 2018.

[66] E. Shelhamer, J. Long, and T. Darrell. "Fully Convolutional Networks for Semantic Segmentation". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 39.4 (Apr. 2017), 640–651. ISSN: 0162-8828. DOI: 10.1109/TPAMI.2016.2572683.

[67] V. Badrinarayanan, A. Kendall, and R. Cipolla. "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39 (2017), pp. 2481–2495.

[68] K. Simonyan and A. Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *CoRR* abs/1409.1556 (2014).

[69] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs." In: *CoRR* abs/1606.00915 (2016).

[70] G. Lin, A. Milan, C. Shen, and I. D. Reid. "RefineNet: Multi-path Refinement Networks for High-Resolution Semantic Segmentation". In: *IEEE CVPR* (2016).

[71] Z. Hengshuang, S. Jianping, Q. Xiaojuan, W. Xiaogang, and J. Jiaya. "Pyramid Scene Parsing Network". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2016.

[72] B. Cheng, L.-C. Chen, Y. Wei, Y. Zhu, Z. Huang, J. Xiong, T. Huang, W.-M. Hwu, and H. Shi. "SPGNet: Semantic Prediction Guidance for Scene Parsing". In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (2019), pp. 5217–5227.

[73] D. Di Mauro, A. Furnari, G. Patanè, S. Battiato, and G. M. Farinella. "SceneAdapt: Scene-based domain adaptation for semantic segmentation using adversarial learning". In: *Pattern Recognition Letters* 136 (2020).

[74] B. Kulis, K. Saenko, and T. Darrell. "What you saw is not what you get: Domain adaptation using asymmetric kernel transforms". In: *CVPR 2011* (2011), pp. 1785–1792.

[75] R. Gopalan, R. Li, and R. Chellappa. “Domain adaptation for object recognition: An unsupervised approach”. In: *2011 International Conference on Computer Vision* (2011), pp. 999–1006.

[76] W. Li, Z. Xu, D. Xu, D. Dai, and L. V. Gool. “Domain Generalization and Adaptation Using Low Rank Exemplar SVMs.” In: *IEEE Trans. Pattern Anal. Mach. Intell.* 40.5 (2018), pp. 1114–1127.

[77] M. Long, Y. Cao, J. Wang, and M. I. Jordan. “Learning Transferable Features with Deep Adaptation Networks”. In: *Proceedings of the 32nd ICML.* 2015.

[78] M. Ghifary, W. B. Kleijn, M. Zhang, D. Balduzzi, and W. Li. “Deep Reconstruction-Classification Networks for Unsupervised Domain Adaptation”. In: *ArXiv* abs/1607.03516 (2016).

[79] O. Sener, H. O. Song, A. Saxena, and S. Savarese. “Learning Transferrable Representations for Unsupervised Domain Adaptation”. In: *NIPS.* 2016.

[80] D. Li, Y. Yang, Y.-Z. Song, and T. M. Hospedales. “Deeper, Broader and Artier Domain Generalization”. In: *2017 IEEE International Conference on Computer Vision (ICCV)* (2017), pp. 5543–5551.

[81] F. M. Carlucci, L. Porzi, B. Caputo, E. Ricci, and S. R. Bulò. “AutoDIAL: Automatic Domain Alignment Layers”. In: *2017 IEEE International Conference on Computer Vision (ICCV)* (2017), pp. 5077–5085.

[82] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio. “Generative Adversarial Nets”. In: *NIPS.* 2014.

[83] J. J. Zhao, M. Mathieu, and Y. LeCun. “Energy-based Generative Adversarial Network”. In: *ArXiv* abs/1609.03126 (2016).

[84] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. “Image-to-Image Translation with Conditional Adversarial Networks”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016), pp. 5967–5976.

[85] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. “Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks”. In: *Computer Vision (ICCV), 2017 IEEE International Conference on.* 2017.

[86] A. Rubhasy, A. A. G. Y. Paramartha, I. Budi, and Z. A. Hasibuan. "Management and retrieval of cultural heritage multimedia collection using ontology". In: *International Conference on Information Technology, Computer, and Electrical Engineering* (2014), pp. 255–259.

[87] P. Kwan, K. Kameyama, J. Gao, and K. Toraichi. "Content-Based Image Retrieval of Cultural Heritage Symbols by Interaction of Visual Perspectives." In: *IJPRAI* 25 (Aug. 2011), pp. 643–673.

[88] D. K. Iakovidis, E. E. Kotsifakos, N. Pelekis, H. Karanikas, I. Kopanakis, T. Mavroudakis, and Y. Theodoridis. "Pattern-Based Retrieval of Cultural Heritage Images". In: 2007.

[89] K. Makantasis, A. Doulamis, N. Doulamis, and M. Ioannides. "In the wild image retrieval and clustering for 3D cultural heritage landmarks reconstruction". In: *Multimedia Tools and Applications* 75.7 (2016), pp. 3593–3629.

[90] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. "Learning realistic human actions from movies." In: *CVPR*. IEEE Computer Society, 2008. ISBN: 978-1-4244-2242-5. URL: http://dblp.uni-trier.de/db/conf/cvpr/cvpr2008.html#LaptevMSR08.

[91] N. Dalal, B. Triggs, and C. Schmid. "Human Detection Using Oriented Histograms of Flow and Appearance". In: *Proceedings of the 9th European Conference on Computer Vision - Volume Part II*. ECCV'06. Graz, Austria: Springer-Verlag, 2006, 428–441. ISBN: 3540338349. DOI: 10.1007/11744047_33. URL: https://doi.org/10.1007/11744047_33.

[92] K. Simonyan and A. Zisserman. "Two-Stream Convolutional Networks for Action Recognition in Videos". In: *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1*. NIPS'14. Montreal, Canada: MIT Press, 2014, 568–576.

[93] C. Feichtenhofer, A. Pinz, and A. Zisserman. "Convolutional Two-Stream Network Fusion for Video Action Recognition". English. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016*. June 2016, pp. 1933–1941.

[94] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool. "Temporal Segment Networks: Towards Good Practices for Deep Action Recognition". In: vol. 9912. Oct. 2016. DOI: 10.1007/978-3-319-46484-8_2.

[95] G. W. Taylor, R. Fergus, Y. LeCun, and C. Bregler. "Convolutional Learning of Spatio-Temporal Features". In: *Proceedings of the 11th European Conference on Computer Vision: Part VI*. ECCV'10. Heraklion, Crete, Greece: Springer-Verlag, 2010, 140–153. ISBN: 3642155669.

[96] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. "Learning Spatiotemporal Features with 3D Convolutional Networks". In: *2015 IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 4489–4497.

[97] J. Carreira and A. Zisserman. "Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), pp. 4724–4733.

[98] G. Varol, I. Laptev, and C. Schmid. "Long-Term Temporal Convolutions for Action Recognition". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.6 (2018), pp. 1510–1517.

[99] B. Zhou, A. Andonian, and A. Torralba. "Temporal Relational Reasoning in Videos". In: *ArXiv* abs/1711.08496 (2018).

[100] C. Feichtenhofer, A. Pinz, and R. P. Wildes. "Spatiotemporal Residual Networks for Video Action Recognition". In: *Proceedings of the 30th International Conference on Neural Information Processing Systems*. NIPS'16. Barcelona, Spain: Curran Associates Inc., 2016, 3476–3484. ISBN: 9781510838819.

[101] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri. "A Closer Look at Spatiotemporal Convolutions for Action Recognition". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 6450–6459.

[102] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy. "Rethinking Spatiotemporal Feature Learning For Video Understanding". In: *CoRR* abs/1712.04851 (2017). arXiv: 1712.04851. URL: http://arxiv.org/abs/1712.04851.

[103] Z. Qiu, T. Yao, and T. Mei. "Learning Spatio-Temporal Representation with Pseudo-3D Residual Networks". In: Oct. 2017, pp. 5534–5542. DOI: 10.1109/ICCV.2017.590.

[104] C. Feichtenhofer, H. Fan, J. Malik, and K. He. "SlowFast Networks for Video Recognition". In: *International Conference on Computer Vision* (2018), pp. 6202–6211.

[105] J. Lin, C. Gan, and S. Han. "TSM: Temporal Shift Module for Efficient Video Understanding". In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 7082–7092.

[106] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, and M. Wray. "Scaling Egocentric Vision: The EPIC-KITCHENS Dataset". In: *European Conference on Computer Vision (ECCV)*. 2018.

[107] T. Nagarajan, C. Feichtenhofer, and K. Grauman. "Grounded Human-Object Interaction Hotspots From Video". In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 8687–8696.

[108] T. Nagarajan, Y. Li, C. Feichtenhofer, and K. Grauman. "EGO-TOPO: Environment Affordances from Egocentric Video". In: *ArXiv* abs/2001.04583 (2020).

[109] K. Fang, T. Wu, D. Yang, S. Savarese, and J. J. Lim. "Demo2Vec: Reasoning Object Affordances from Online Videos". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 2139–2147.

[110] M. Cai, K. M. Kitani, and Y. Sato. "Understanding Hand-Object Manipulation with Grasp Types and Object Attributes". In: *Robotics: Science and Systems*. 2016.

[111] S. Bambach, S. Lee, D. J. Crandall, and C. Yu. "Lending A Hand: Detecting Hands and Recognizing Activities in Complex Egocentric Interactions". In: *2015 IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 1949–1957.

[112] D. Shan, J. Geng, M. Shu, and D. Fouhey. "Understanding Human Hands in Contact at Internet Scale". In: 2020.

[113] H. Li, W.-S. Zheng, Y. Tao, H. Hu, and J.-H. Lai. “Adaptive Interaction Modeling via Graph Operations Search”. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020.

[114] R. Goyal, S. E. Kahou, V. Michalski, J. Materzynska, S. Westphal, H. Kim, V. Haenel, I. Fruend, P. Yianilos, M. Mueller-Freitag, F. Hoppe, C. Thurau, I. Bax, and R. Memisevic. “The “Something Something” Video Database for Learning and Evaluating Visual Common Sense”. In: *2017 IEEE International Conference on Computer Vision (ICCV)*. 2017, pp. 5843–5851.

[115] Y. Chao, Z. Wang, Y. He, J. Wang, and J. Deng. “HICO: A Benchmark for Recognizing Human-Object Interactions in Images”. In: *2015 IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 1017–1025.

[116] B. G. Fabian Caba Heilbron Victor Escorcia and J. C. Niebles. “ActivityNet: A Large-Scale Video Benchmark for Human Activity Understanding”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 961–970.

[117] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, A. Natsev, M. Suleyman, and A. Zisserman. “The Kinetics Human Action Video Dataset”. In: *ArXiv* abs/1705.06950 (2017).

[118] J. Carreira, E. Noland, C. Hillier, and A. Zisserman. “A Short Note on the Kinetics-700 Human Action Dataset”. In: *ArXiv* abs/1907.06987 (2019).

[119] D. Damen, H. Doughty, G. M. Farinella, A. Furnari, J. Ma, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, and M. Wray. “Rescaling Egocentric Vision”. In: *CoRR* abs/2006.13256 (2020).

[120] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, and M. Wray. “The EPIC-KITCHENS Dataset: Collection, Challenges and Baselines”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2020).

[121] Y. Li, M. Liu, and J. M. Rehg. “In the Eye of Beholder: Joint Learning of Gaze and Actions in First Person Video”. In: *ECCV*. 2018.

[122] F. de la Torre, J. K. Hodgins, J. Montano, and S. Valcarcel. "Detailed Human Data Acquisition of Kitchen Activities: the CMU-Multimodal Activity Database (CMU-MMAC)". In: *CHI 2009 Workshop. Developing Shared Home Behavior Datasets to Advance HCI and Ubiquitous Computing Research.* 2009.

[123] H. Pirsiavash and D. Ramanan. "Detecting activities of daily living in first-person camera views." In: *CVPR*. IEEE Computer Society, 2012, pp. 2847–2854. ISBN: 978-1-4673-1226-4. URL: http://dblp.uni-trier.de/db/conf/cvpr/cvpr2012.html#PirsiavashR12.

[124] Y. Tang, Y. Tian, J. Lu, J. Feng, and J. Zhou. "Action recognition in RGB-D egocentric videos". In: *2017 IEEE International Conference on Image Processing (ICIP)*. 2017, pp. 3410–3414.

[125] W. Li, Z. Zhang, and Z. Liu. "Action recognition based on a bag of 3D points". In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops* (2010), pp. 9–14.

[126] J. Wang, Z. Liu, Y. Wu, and J. Yuan. "Mining actionlet ensemble for action recognition with depth cameras". In: *2012 IEEE Conference on Computer Vision and Pattern Recognition* (2012), pp. 1290–1297.

[127] J. Sung, C. Ponce, B. Selman, and A. Saxena. "Human Activity Detection from RGBD Images". In: *Proceedings of the 16th AAAI Conference on Plan, Activity, and Intent Recognition.* AAAIWS'11-16. AAAI Press, 2011, 47–55.

[128] H. S. Koppula, R. Gupta, and A. Saxena. "Learning human activities and object affordances from RGB-D videos". In: *The International Journal of Robotics Research* 32.8 (2013), pp. 951–970. DOI: 10.1177/0278364913478446. eprint: https://doi.org/10.1177/0278364913478446. URL: https://doi.org/10.1177/0278364913478446.

[129] C. Chen, R. Jafari, and N. Kehtarnavaz. "UTD-MHAD: A multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor". In: *2015 IEEE International Conference on Image Processing (ICIP)*. 2015, pp. 168–172. DOI: 10.1109/ICIP.2015.7350781.

[130] H. Rahmani, A. Mahmood, D. Huynh, and A. Mian. "Histogram of Oriented Principal Components for Cross-View Action Recognition". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38 (2016), pp. 2430–2443.

[131] J. Hu, W. Zheng, J. Lai, and J. Zhang. "Jointly Learning Heterogeneous Features for RGB-D Activity Recognition". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.11 (2017), pp. 2186–2200. DOI: 10.1109/TPAMI.2016.2640292.

[132] J. Liu, A. Shahroudy, M. Perez, G. Wang, L. yu Duan, and A. Kot. "NTU RGB+D 120: A Large-Scale Benchmark for 3D Human Activity Understanding". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42 (2020), pp. 2684–2701.

[133] Z. Zhang. "Microsoft Kinect Sensor and Its Effect". In: *IEEE MultiMedia* 19.2 (Apr. 2012), 4–10. ISSN: 1070-986X. DOI: 10.1109/MMUL.2012.24. URL: https://doi.org/10.1109/MMUL.2012.24.

[134] G. Garcia-Hernando, S. Yuan, S. Baek, and T.-K. Kim. "First-Person Hand Action Benchmark with RGB-D Videos and 3D Hand Pose Annotations". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018), pp. 409–419.

[135] M. Moghimi, P. Azagra, L. Montesano, A. C. Murillo, and S. Belongie. "Experiments on an RGB-D Wearable Vision System for Egocentric Activity Recognition". In: *2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2014, pp. 611–617. DOI: 10.1109/CVPRW.2014.94.

[136] S. Song, V. Chandrasekhar, B. Mandal, L. Li, J. Lim, G. S. Babu, P. P. San, and N. Cheung. "Multimodal Multi-Stream Deep Learning for Egocentric Activity Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2016, pp. 378–385. DOI: 10.1109/CVPRW.2016.54.

[137] G. Rogez, J. S. Supancic, and D. Ramanan. "Understanding Everyday Hands in Action from RGB-D Images". In: *2015 IEEE International Conference on Computer Vision (ICCV)*. 2015, pp. 3889–3897. DOI: 10.1109/ICCV.2015.443.

[138] R. Kothari, Z. Yang, C. Kanan, R. J. Bailey, J. Pelz, and G. J. Diaz. “Gaze-in-wild: A dataset for studying eye and head coordination in everyday activities”. In: *Scientific Reports* 10 (2020).

[139] A. Furnari, S. Battiato, K. Grauman, and G. Farinella. “Next-active-object prediction from egocentric videos”. In: *J. Vis. Commun. Image Represent.* 49 (2017), pp. 401–411.

[140] M. Liu, S. Tang, Y. Li, and J. M. Rehg. “Forecasting Human-Object Interaction: Joint Prediction of Motor Attention and Actions in First Person Video”. In: *Computer Vision – ECCV 2020.* Ed. by A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm. Cham: Springer International Publishing, 2020, pp. 704–721. ISBN: 978-3-030-58452-8.

[141] J. Jiang, Z. Nan, H. Chen, S. Chen, and N. Zheng. “Predicting short-term next-active-object through visual attention and hand position”. In: *Neurocomputing* 433 (2021), pp. 212–222. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2020.12.069. URL: https://www.sciencedirect.com/science/article/pii/S092523122031969X.

[142] C. Fan, J. Lee, and M. Ryoo. “Forecasting Hand and Object Locations in Future Frames”. In: *ArXiv* abs/1705.07328 (2017).

[143] E. Dessalene, C. Devaraj, M. Maynord, C. Fermuller, and Y. Aloimonos. “Forecasting Action through Contact Representations from First Person Video”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021), pp. 1–1. DOI: 10.1109/TPAMI.2021.3055233.

[144] F. Ragusa, A. Furnari, S. Battiato, G. Signorello, and G. M. Farinella. “Egocentric Visitors Localization in Cultural Sites”. In: *ACM Journal on Computing and Cultural Heritage* (2018).

[145] F. Ragusa, A. Furnari, S. Battiato, G. Signorello, and G. M. Farinella. “Egocentric Point of Interest Recognition in Cultural Sites”. In: *International Conference on Computer Vision Theory and Applications (VISAPP).* 2019. URL: http://iplab.dmi.unict.it/VEDI_POIs/.

[146] F. Ragusa, A. Furnari, S. Battiato, G. Signorello, and G. M. Farinella. “EGO-CH: Dataset and Fundamental Tasks for Visitors Behavioral Understanding using Egocentric Vision”. In: *Pattern Recognition Letters* (2020).

[147] F. Ragusa, D. D. Mauro, A. Palermo, A. Furnari, and G. M. Farinella. "Semantic Object Segmentation in Cultural Sites using Real and Synthetic Data". In: *International Conference on Pattern Recognition (ICPR).* 2020.

[148] A. Dutta, A. Gupta, and A. Zissermann. *VGG Image Annotator (VIA).* http://www.robots.ox.ac.uk/ vgg/software/via/. 2016.

[149] B. O. Community. *Blender - a 3D modelling and rendering package.* Blender Foundation, 2018. URL: http://www.blender.org.

[150] S. Orlando, A. Furnari, and G. M. Farinella. "Egocentric Visitor Localization and Artwork Detection inCultural Sites Using Synthetic Data". In: *Pattern Recognition Letters* (2020).

[151] C. M. Bishop. *Pattern recognition and Machine Learning.* Springer, 2006.

[152] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. "Caffe: Convolutional architecture for fast feature embedding". In: *Proceedings of the 22nd ACM international conference on Multimedia.* ACM. 2014, pp. 675–678.

[153] I. Koprinska and S. Carrato. "Temporal video segmentation: A survey". In: *Signal Processing: Image Communication.* 2001, pp. 477–500.

[154] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger. "Densely connected convolutional networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* 2017.

[155] G. M. Farinella, G. Signorello, S. Battiato, A. Furnari, F. Ragusa, R. Leonardi, E. Ragusa, E. Scuderi, A. Lopes, L. Santo, and M. Samarotto. "VEDI: Vision Exploitation for Data Interpretation". In: *Image Analysis and Processing – ICIAP 2019.* Ed. by E. Ricci, S. Rota Bulò, C. Snoek, O. Lanz, S. Messelodi, and N. Sebe. Cham: Springer International Publishing, 2019, pp. 753–763. ISBN: 978-3-030-30645-8.

[156] F. L. M. Milotta, A. Furnari, S. Battiato, M. D. Salvo, G. Signorello, and G. M. Farinella. "Visitors Localization in Natural Sites Exploiting EgoVision and GPS". In: *International Conference on Computer Vision Theory and Applications (VISAPP).* 2019.

[157] F. Ragusa, L. Guarnera, A. Furnari, S. Battiato, G. Signorello, and G. M. Farinella. “Localization of Visitors for Cultural Sites Management”. In: *International Joint Conference on e-Business and Telecommunications - Volume 2: ICETE*. 2018, pp. 407–413.

[158] *Epson Moverio BT 300*. https://www.epson.eu/products/see-through-mobile-viewer/moverio-bt-300.

[159] *Microsoft Hololens 2*. https://www.microsoft.com/en-us/hololens.

[160] *Vuzix Blade*. https://www.vuzix.com/products/blade-smart-glasses.

[161] S. Colombo, Y. Lim, and F. Casalegno. “Deep Vision Shield: Assessing the Use of HMD and Wearable Sensors in a Smart Safety Device”. In: *Proceedings of the 12th ACM International Conference on PErvasive Technologies Related to Assistive Environments*. PETRA ’19. Rhodes, Greece: Association for Computing Machinery, 2019, 402–410. ISBN: 9781450362320. DOI: 10.1145/3316782.3322754. URL: https://doi.org/10.1145/3316782.3322754.

[162] B. Soran, A. Farhadi, and L. Shapiro. “Generating Notifications for Missing Actions: Don’t Forget to Turn the Lights Off!” In: Dec. 2015, pp. 4669–4677. DOI: 10.1109/ICCV.2015.530.

[163] G. Cotugno, D. Turchi, D. Russell, and G. Deacon. “SecondHands: A Collaborative Maintenance Robot for Automated Warehouses. Implications for the Industry and the Workforce”. In: *Inclusive Robotics for a Better Society*. Ed. by J. L. Pons. Cham: Springer International Publishing, 2020, pp. 195–200. ISBN: 978-3-030-24074-5.

[164] F. Ragusa, A. Furnari, S. Livatino, and G. M. Farinella. “The MECCANO Dataset: Understanding Human-Object Interactions from Egocentric Videos in an Industrial-like Domain”. In: *IEEE Winter Conference on Application of Computer Vision (WACV)*. 2021. eprint: 2010.05654 (cs.CV). URL: https://iplab.dmi.unict.it/MECCANO.

[165] T. L. A. Nijmegen: Max Planck Institute for Psycholinguistics. “ELAN (Version 5.9) [Computer software]”. In: (2020). URL: "https://archive.mpi.nl/tla/elan.

[166] A. Dutta and A. Zisserman. "The VIA Annotation Software for Images, Audio and Video". In: *Proceedings of the 27th ACM International Conference on Multimedia*. MM '19. Nice, France: ACM, 2019. ISBN: 978-1-4503-6889-6/19/10. DOI: 10.1145/3343031.3350535. URL: https://doi.org/10.1145/3343031.3350535.

[167] H. Fan, Y. Li, B. Xiong, W.-Y. Lo, and C. Feichtenhofer. *PySlowFast*. https://github.com/facebookresearch/slowfast. 2020.

[168] K. He, X. Zhang, S. Ren, and J. Sun. "Deep Residual Learning for Image Recognition". In: *arXiv preprint arXiv:1512.03385* (2015).

[169] G. A. Sigurdsson, A. Gupta, C. Schmid, A. Farhadi, and A. Karteek. "Actor and Observer: Joint Modeling of First and Third-Person Videos". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2018), pp. 7396–7404.

[170] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick. *Detectron2*. https://github.com/facebookresearch/detectron2. 2019.

[171] M. Everingham, S. M. Eslami, L. Gool, C. K. Williams, J. Winn, and A. Zisserman. "The Pascal Visual Object Classes Challenge: A Retrospective". In: *Int. J. Comput. Vision* 111.1 (Jan. 2015), 98–136. ISSN: 0920-5691. DOI: 10.1007/s11263-014-0733-5. URL: https://doi.org/10.1007/s11263-014-0733-5.

[172] D. Misra, T. Nalamada, A. U. Arasanipalai, and Q. Hou. "Rotate to Attend: Convolutional Triplet Attention Module". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2021, pp. 3139–3148.

La borsa di dottorato è stata cofinanziata con risorse del
Programma Operativo Nazionale Ricerca e Innovazione 2014-2020 (CCI 2014IT16M2OP005),
Fondo Sociale Europeo, Azione I.1 "Dottorati Innovativi con caratterizzazione Industriale"

UNIONE EUROPEA
Fondo Sociale Europeo

Ministero dell'Università e della Ricerca

PON
RICERCA E INNOVAZIONE
2014 - 2020