A Microservices Architecture based on a Deep-learning Approach for an Innovative Fruition of Art and Cultural Heritage

Ilaria Sergi, Marco Leo, Pierluigi Carcagnì, Marco La Franca, Cosimo Distante, and Luigi Patrono

***Abstract*— Technological innovations have resulted in a digital transformation in a variety of fields, including culture and tourism. We propose an innovative and personalized solution to benefit art and cultural heritage in indoor and outdoor environments by combining Internet of Things-enabled technologies and deep learning-based approaches. A recent Convolutional Neural Network (CNN) architecture to jointly perform local feature detection and description has been adapted and exploited for the first time for image matching in the cultural heritage application context. The performance validation of the proposed system shows that the proposed modular architecture ensures a very low error rate and excellent response time up to 2000 user visits in 700 seconds. The validation of the computer vision module shows as the proposed CNN based feature extraction approach improves image matching performance, especially in poorly textured object areas reaching a F1-Score of 0.9907 (against the 0.9679 obtained by traditional gradient based approaches) on the challenging dataset of images taken from 4 different historical sites and a F1-Score of 0.9807 (against the 0.9798 obtained by traditional approaches) on a public benchmark dataset of artworks.**

***Index Terms*—deep learning, image matching, internet of things, microservices, performance validation.**

I. INTRODUCTION

THE use of Information and Communication Technologies (ICT) for management, protection and fruition of cultural heritage is gradually spreading in every country in the world.

Art and cultural venues inevitably need to use ICT for immersive and augmented tours and for greater viewer engagement. The combined use of these technologies and Machine Learning models, especially the development of solutions based on Deep Learning, can automatically identify a visual content and describe it in natural language. This represents a real strength in achieving ambitious goals, including: (i) creation of new adaptive and innovative research and acquisition forms of cultural heritage; (ii) creation of new and attractive tools to bring young people closer to art and cultural heritage; (iii) creation of new cultural tools not only for experts but also for citizens and tourists, thereby enhancing the artistic, historical, and cultural heritage of the territory. In most art and culture sites (museums, galleries, etc.) the fruition of content is mainly entrusted to audio guides that cannot provide visitors with a unique and personalized experience. This can be achieved by introducing innovative architectures that use Internet of Things (IoT)-enabling technologies [1], Cloud Computing services as well as advanced Computer Vision techniques to distinguish the delivered contents. Indeed, IoT technologies are frequently combined with other technologies or services in the creation of intelligent environments to address specific challenges such as scalability and high performance, which can be obtained through Cloud Computing, intelligent data processing with artificial intelligence techniques, or latency-sensitive monitoring and privacy-aware learning through Edge Computing [2]. Furthermore, through the last decade, the IoT definition, technologies and scope have been evolving embracing very different challenges and reference application scenarios [3] and among these the tourism sector occupies a relevant place.

From this perspective, this work proposes an innovative and complex architecture that combines IoT and computer vision techniques to create a smart system for the personalized fruition of art and cultural content. In particular, the software solution is based on a microservices architecture able to offer flexibility in choosing technologies (each microservice can be written using a different technology) and allow painless addition of new components to the system or scale services separately from one another. A key feature of the proposed framework consists of understanding the relationships among multiple visual targets in order to identify them in different images acquired from

Manuscript received March 4, 2022; revised March 27, 2022. Date of publication April 30, 2022. Date of current version April 30, 2022.

I. Sergi and L. Patrono are with the Department of Engineering for Innovation, University of Salento, Via per Monteroni snc, Lecce, 73100, Italy (emails: ilaria.sergi@unisalento.it; luigi.patrono@unisalento.it).

M. Leo, P. Carcagnì, and C. Distante are with the Institute of Applied Sciences and Intelligent Systems, National Research Council of Italy, Via per Monteroni snc, Lecce, 73100, Italy (emails: marco.leo@isasi.cnr.it; pierluigi.carcagni@cnr.it; cosimo.distante@cnr.it).

M. La Franca is with Cloudtec s.r.l., Via Trapani 1/D, Palermo, 90141, Italy (email: marco.lafranca@gmail.com).

Digital Object Identifier (DOI): 10.24138/jcomss-2022-0001

different devices, in different light conditions and under affine transformations, and even under non-linear deformations. This problem is referred in literature as image matching, also known as image registration or correspondence, and it plays a crucial role in various fields, including computer vision, pattern recognition, image analysis, security, and remote sensing.

Image matching has a history of more than 50 years, with the first experiments performed with analog procedures for cartographic and mapping purposes. Over the past decades, a growing amount and diversity of methods have been proposed for image matching, particularly with the development of deep learning techniques. However, there are several open questions about the choice of suitable methods for specific applications in terms of accuracy, robustness, and efficiency.

In this paper a deep learning-based approach for matching image content related to artworks (monuments, pictures, and statues) has been introduced.

Indeed, a recently introduced CNN architecture [3], able to perform both local feature detection and description has been adapted and exploited, by our knowledge for the first time, for image matching in the cultural heritage application context.

It can model the local shape for stronger geometric invariance and accurate localization of the keypoints.

It resorts to stacked deformable convolutional networks for a progressive shape modelling leading to a dense local transformation prediction by which leads a better handling of geometric variations is achieved.

On the other hand, it takes advantage of the inherent feature hierarchy to restore spatial resolution and low-level details for accurate keypoint localization. This is achieved by leveraging the inherent pyramidal feature hierarchy of CNN and combining detections from multiple feature levels.

Finally, it uses a specific peak measurement to relate feature responses at different scales and this way it derives more indicative detection scores which helps to better preserve the low-level structures such as corners or edges.

Comparison with state-of-the-art results is reported that demonstrate the effectiveness of the proposed image matching approach in the cultural heritage application field. Aside from the aforementioned image matching strategy, a significant paper contribution is the use of a modular architectural style in which the proposed software project is divided into small, independent, loosely coupled, and separately deployable service parts, referred to as microservices, which communicate with one another via simple Application Programming Interfaces (APIs). The developed architecture ensures improved fault isolation, smaller and faster deployment and scalability, which is one of the many benefits of using this microservice. Indeed, because the developed services are distinct, the most critical ones can be easily scaled at the appropriate times while also saving money.

To summarize, the following are the main contributions of the paper: (i) introduction of a deep learning-based approach for matching image content related to artworks; (ii) comparison of the proposed deep-learning approach with traditional matching approaches; (iii) design and implementation of a modular architecture for personalized fruition of art and cultural content; and (iv) functional and performance validation of the proposed architecture.

The rest of the paper is structured as follows. In section II a state-of-the-art about existing ICT solutions designed to enhance art and cultural heritage is presented, with a particular focus on solutions exploiting Machine Learning approaches. Section III summarizes the main requirements for innovative systems for personalized fruition of art and cultural content. Section IV introduces the proposed system architecture and describes each system component. A functional validation of the proposed system is presented in section V whereas section VI discusses main results of the performance validation performed on both the computer vision module and the proposed software platform. Finally, conclusions are drawn in section VII.

II. Related Work

In recent years, the introduction of ICT in the tourism industry, especially in the context of enhancing cultural heritage, has been the focus of many scientific studies [5]. Innovative solutions for the smart fruition of cultural information is also an interesting and deeply explored research field. Extensive research has been conducted in this field and remarkable results have been achieved [8], [9]. In particular, technologies, protocols, and devices enabling the IoT (such as Bluetooth Low Energy, sensors, embedded devices, mobile devices, and Cloud Computing services) can be exploited to attract new audiences to art and cultural venues by facilitating access to the cultural heritage and by improving the quality of the visit with opportunities for personal fruition with entertaining and interactive learning. Furthermore, the combined use of IoT technologies and Machine Learning techniques facilitates the creation of solutions able to enhance a user's visiting experience and improve the process of transmitting knowledge of a cultural site [10].

Beyond the application scope, there are a set of works that are related to this by the involved methodological approaches. In a broader sense, the computer vision component of the framework exploits findings related to the application of machine learning to cultural heritage. This is a spreading research area whose growing attention from many research groups is clear throughout the literature [13]. In a narrower sense, the main topic of the aforementioned computer vision component is image matching. Image matching is one of the most researched topics in computer vision and there are plenty of works on that. However, it remains an important and challenging problem in computer vision. Several categorizations have been proposed. Some of them rely on potential variations between the two images: they distinguish if the two images are assumed to be of the same object varied by motion, or the same scene with affine transformations. This last may occur in dynamic scenes in video sequences or variations can arise from the appearance of different object instances, such as cars with various shapes and colors, people with different poses and clothing, and animals of different species [14]. Other categorizations are based on the approaches used to address the image matching. *Area-based methods* take

into account image pixel intensity without attempting to detect any salient image structure whereas *feature-based methods* extract distinctive structure from an image and compare them. Feature-based methods are often more efficient, robust, and accurate since they can better handle geometrical deformations and radiometric differences as well [15]. On the other hand, they are dependent on feature detection, description, and comparison in metric space which are usually three of the most challenging tasks which have to be defined in the different application fields. Detected features represent specific semantic structures in an image or the real world and can refer to corners, blobs, lines/edges, morphological regions. However, the most popular features that are used for matching are the points (a.k.a. keypoints or interest points) since they are easy to extract and define with a simplified form. A crucial role in this arena is played by SIFT [16] (and its modified versions, e.g. SURF [31]) that still represents a reference for any new image matching algorithm [15]. In recent years, data-driven learning-based methods have achieved significant progress in general visual pattern recognition tasks and have also been applied to image feature detection. This pipeline can be roughly classified into the use of classical learning and deep learning. CNN-based approaches construct a response map to search for the interest points in a supervised, self-supervised or unsupervised manner. The task is often converted into a regression problem that can be trained in a differentiable way under the transformation and imaging condition invariance constraints. During the past few years, the joint learning of local feature detectors and descriptors has gained increasing popularity, with promising results achieved by novel deep architectures in real applications [17], [18]. Unfortunately, efficient local shape estimation and keypoint localization accuracy require additional investigation and for this purpose, new CNN network architectures are continuously introduced and tested [3]. A comprehensive and systematic review and analysis for those classical and latest techniques can be found in [19].

III. System Requirements

Starting from the study of existing works dealing with systems based on IoT technologies and image processing techniques for the fruition of cultural heritage, the main requirements of an innovative system for the personalized fruition of art and cultural content have been pointed out and they are reported here.

The proposed system should be able to enhance the experience of visitors in indoor and outdoor arts and cultural venues by providing them with audio descriptions of the observed artwork (paintings, buildings, monuments, sculptures, etc.) based on user language and profile (adult, child, art expert, etc.). Tourists must be able to access the services provided by the system through mobile devices. In addition, the system must allow the user to enter a minimum set of information (e.g., date of birth, gender, nationality) that can classify the same user to a specific profile in the system. Image processing techniques must be used to identify a work of art starting from the photo taken by the tourist via a mobile device. In addition to photos, the system can also receive the user's location (indoor or outdoor) to speed up the process of identifying the photographed artwork. Over time, the system must be sustainable. Therefore, an effective mechanism must be provided to allow authorized users to create new content and modify existing content. Appropriate authorization and authentication mechanisms must be provided to allow users with different roles to guarantee the different functions offered by the platform. In particular, the following roles must be envisioned:

- *end user*. Generic user (e.g., tourist) that visualizes information (POIs description, images, maps) loaded into the system;
- *editor*. The user that verifies the content posted by other authorized users (contributors, translators) before making them public. In addition, the editor assigns content that needs to be translated into other languages to the translators;
- *contributor* (e.g., tourist guide). The user that enters new content into the platform, such as new POI or the description of the artwork for a specific user profile (adults, children, art experts, etc.);
- *translator*. The user that handles the translation of textual content. Translators are pre-uploaded into the platform so that editors can quickly select users to assign translations based on their language skills;
- *administrator*. This user manages any content on the platform and, in addition, manages user registrations, assigning a role to each new user (editor, contributor, translator).

The authentication must be required for all listed user roles. The description of the artwork will be entered in text form and stored into the platform. The mobile application used by tourists will convert text content into audio content through a specific module implemented. As mentioned earlier, the mobile application must provide user authentication and profiling mechanisms to allow the system to send content based on the user profile. The back-end server will have to best manage the stored data, be modular and scalable, so it can support future system expansion. The web application must clearly display information to users to facilitate the content management.

IV. Proposed Architecture

The proposed system architecture is reported in Fig. 1. It consists of three main macro-components: (i) the mobile App, (ii) the Web App, and (iii) the Cloud Server.

A. Mobile App

The mobile application allows tourists visiting indoor (museums, art galleries, etc.) or outdoor (historical center of the city, historical or archaeological routes, etc.) art and cultural places to use their mobile device (smartphone or tablet) to take photos of artwork (paintings, sculptures, church facades, etc.) or provide details of them, and receive descriptions in audio format. The mobile app will also capture the tourist's indoor location (obtained via BLE beacon infrastructure) or outdoor location (obtained via GPS positioning), if this type of information is available, to minimize the number of images to be compared for faster response. In the registration phase, the mobile application will ask the tourist to fill out a short profile form so that the system can classify the user into one of the

profiles provided by the system (adult, child, expert). The mobile application has been developed using the Ionic open-source framework [20], which makes full use of Web technology to create a hybrid mobile application with a user interface (UI) that is very similar to native applications.

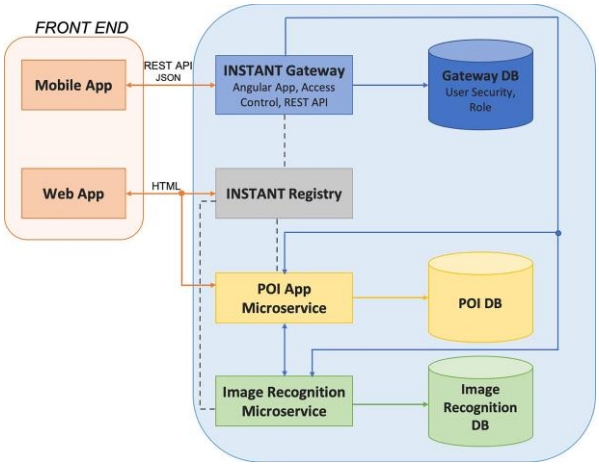

Fig. 1. Block diagram of the INSTANT system architecture.

B. Web App

The front-end web application is dedicated to content management and allows each user to perform specific actions based on the role played in the platform (guest, editor, contributor, translator, administrator). In particular, it allows the management of users and POIs and all information related to them. Various technologies have contributed to the creation of the web application: HTML 5, Bootstrap [21], Angular 7 [22], responsive web design, Websocket [23], Angular Translate [24]. The application uses a responsive layout and can also be accessed from a mobile browser. The content is available in five languages: Italian, English, French, Spanish and German. For security management, the Angular front-end uses standard solutions to provide these services:

- form-based authentication, where the user provides his own credentials (username and password) in order to log into the system;
- when the user successfully logs in with his credentials, the system will return a JSON Web Token (JWT) [25], save it locally and use it for access protected resources;
- all data are exchanged in JSON format.

C. Cloud Server

The Cloud Server platform uses an innovative microservices architecture, which includes the following modules: the INSTANT Gateway, the INSTANT Registry, the POI App Microservice, and the Image Recognition Microservice.

Microservices represent specific application modules dedicated to managing specific macro-functionalities (such as POI management, image recognition). From a data point of view, they are isolated from each other. In fact, each microservice accesses an independent database. If necessary, a microservice can interact with the gateway or another microservice through HTTP REST APIs. By inserting a JWT token in the header of each call, security can be guaranteed.

1) INSTANT Gateway

The gateway module exposes the REST APIs for interacting with the mobile application. It contains:

- the Angular application for the management of all microservices;
- the HTTP routing module and the Load Balancing module;
- the security manager (users and roles);
- the REST APIs for access from the mobile application;
- the REST APIs documentation (via Swagger [26], an interesting open-source project fully dedicated to the REST APIs documentation).

The gateway has been implemented using Spring Boot, which is an open source framework for developing applications on the Java platform. The user interface is developed using the Angular framework. Instant Gateway manages users and application roles and provides interfaces for all microservices. It connects with microservices to read or save data (such as POI). It accesses the MySQL database [27] to store users and application roles.

2) INSTANT Registry

It represents the backend "orchestrator" module. It monitors all other services (gateway and microservices) and includes the following main functions:

- It is a Eureka server, that is, it acts as a "discovery server" for applications and allows all applications to perform routing management, load balancing, and scalability.
- It is a Spring Cloud Config server that provides runtime configuration for all applications in the backend.
- It is a management server that includes dashboards for monitoring and managing applications.

All its functions can be used through a dedicated Angular application. The registry is the only module that does not need to access the database.

3) POI App Microservice

The Poi App microservice allows management of POIs and other related entities (content, attachments, categories). This module provides create, read, update, and delete (CRUD) operations for POI entities and associated business logic. It manages the translation of content, attachments, and POI categories. It also allows management of workflows so that new content can be approved by the administrator or editor. It is a Java Spring Boot project and accesses a dedicated database to store POIs and all related entities.

4) Image Recognition Microservice

It is responsible for identifying POIs from images. It finds a possible matching between images taken from the user and representative images previously enrolled. The image matching leverages a complementary computer vision-based module exploiting deep learning models in convolutional architectures.

Inspired by a previous network architecture [3], the key elements in the deep model exploited for learning local features are deformable convolutional networks (DCN) [28] that predict and apply dense spatial transformation, and D2-Net [17] that

jointly learns keypoint detector and descriptor.

Deformable Convolutional Networks introduce two new modules to enhance the transformation modeling capability of CNNs, namely, deformable convolution and deformable RoI pooling. Both are based on the idea of augmenting the spatial sampling locations in the modules with additional offsets and learning the offsets from the target tasks, without additional supervision. The new modules can readily replace their plain counterparts in existing CNNs and can be easily trained end-to-end by standard backpropagation, giving rise to deformable convolutional networks.

The implementation starts from AffNet [29], and it constructs a network to predict three scalars to model the scaling factor and rotation (cosine and sine, which are then used to compute an angle). No geometrical constraint is introduced. D2-Net then extracts feature descriptions and detections by applying channel-wise L2-normalization to obtain dense feature descriptors, while the feature detections are derived from the local score and the channel-wise score. Given a feature hierarchy consisting of feature maps at different levels the above-mentioned detection is applied at each level to get a set of score maps. Next, each score map is upsampled to have the same spatial resolution as input and finally combined by taking the weighted sum.

V. Functional Validation of the INSTANT System

The functional validation of a system allows us to check its correct operation and to verify that the system satisfies all the specified requirements.

The functional validation of the INSTANT system has been carried out by analyzing four main use cases:

1. The Editor accesses the Web App and adds a new POI.
2. The Editor accesses the Web App and assigns tasks to Contributor and Translator for the POI management.
3. The Contributor and the Translator take in charge all tasks assigned by the Editor.
4. The end user accesses the Web App and visualize the new content.

A. The Editor Adds a New POI

The Editor uses his security credentials to log in to the application. Then s/he accesses the POI management screen (POI management menu -> POI), and then clicks the "New POI" button (Fig. 2). The Editor enters the required data (name, address) and clicks the "Save" button to insert the POI on the platform. Now, the new POI will be visible in the POI list accessible via the “POI management” menu → “POI”. This POI is in the "draft" state, so it is ready to be assigned to a Contributor or a Translator to provide content (POI description for different user profiles and in different languages).

B. The Editor Assigns Tasks to Contributor and Translator

The editor uses his/her security credentials to log in to the application. Then, s/he accesses the "POI Management" screen (POI Management Menu → Task) and clicks the "New Task" button. The editor inserts the required information (s/he selects the POI from the POIs registered in the system, enters the title and the deadline of the task). Then, s/he assigns tasks to users with the role of contributor or translator, and defines the priority of the task (high, medium, low). The system will send emails to users who have assigned tasks. Fig. 3 shows a screenshot where new tasks are assigned to a Contributor.

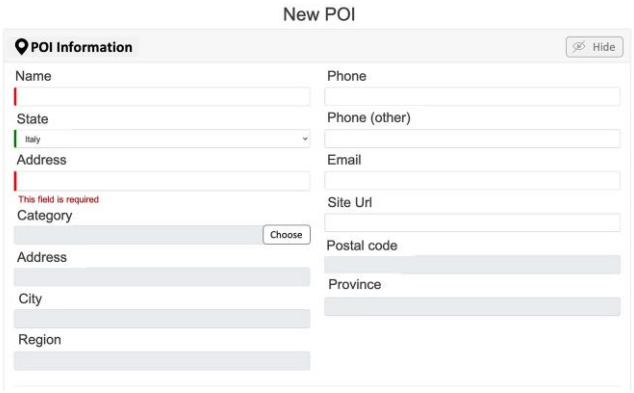

Fig. 2. The Editor accesses the Web App and adds a new POI.

C. The Editor Assigns Tasks to Contributor and Translator

The editor uses his/her security credentials to log in to the application. Then, s/he accesses the "POI Management" screen (POI Management Menu → Task) and clicks the "New Task" button. The editor inserts the required information (s/he selects the POI from the POIs registered in the system, enters the title and the deadline of the task). Then, s/he assigns tasks to users with the role of contributor or translator, and defines the priority of the task (high, medium, low). The system will send emails to users who have assigned tasks. Fig. 3 shows a screenshot where new tasks are assigned to a Contributor.

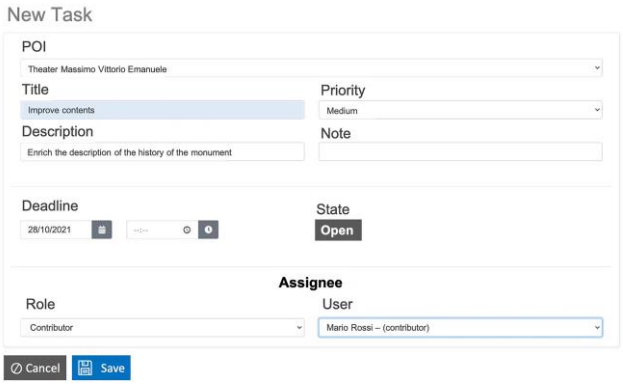

Fig. 3. The Editor accesses the Web App and assigns tasks to a Contributor for the POI management.

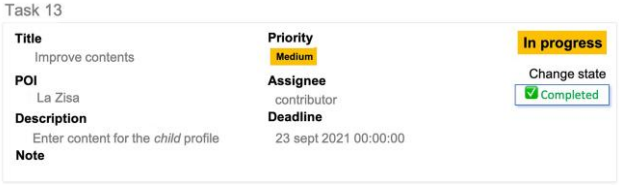

Fig. 4. The Contributor takes in charge the tasks assigned by the Editor.

D. Contributor and Translator Take in Charge Assigned Tasks

After Contributor and Translator log in to the Web App, they can access the POI management screen (POI management menu

→ tasks), select one of the assigned tasks to read detailed information, and set the status of the task to "In progress" (Fig. 4). After the task is completed, the contributor or translator will set the "Completed" status and insert a short description of the activity in the "Note" section. The Editor is notified of all status changes. Once a Contributor or Translator has completed the requested task, the Editor will evaluate the completed work. If the work is deemed unsatisfactory, the Editor can reopen and reassign the task, otherwise the Editor will set the status to "Approved" to approve the new content.

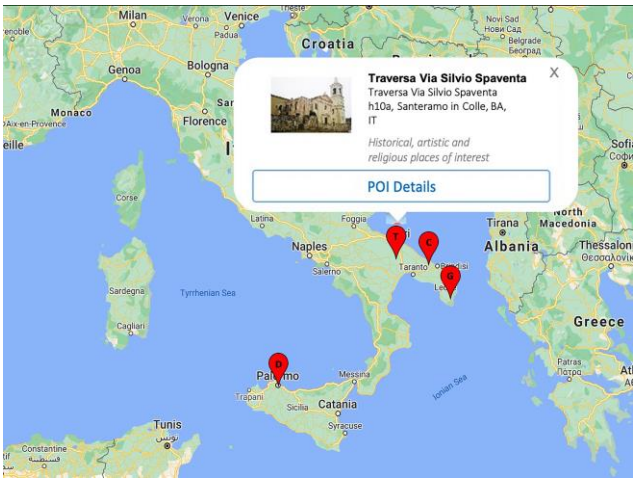

Fig. 5. The end user accesses the Web App and visualizes the new content on the map.

E. The End User Accesses the Web App and Visualizes the New Content

The end user accesses the Web App and displays all available POIs on the map. By clicking the mark, s/he will be able to see the main information of the selected POI (Fig.5). By clicking the "Details" button, the user can access POI details and view all information based on his/her profile.

VI. Results and Discussion

In this section, the experimental results related to the performance validation carried out on the computer vision module and the INSTANT system are reported and discussed. Detailed information about the considered settings is also provided.

A. Performance of the Computer Vision Module

The introduced computer vision module has been tested in two different experimental phases. In both phases the proposed CNN approach has been compared with the classical reference matching scheme based on SIFT [16].

The choice of SIFT as reference algorithm was made taking into consideration comparative studies (e.g., the one in [32]) that proved SIFT to be more accurate and efficient (on global image matching) than other feature matching algorithms. Besides, as a secondary motivation, SIFT implementations can be easily reproduced or community approved versions can be found on the web making comparisons free from implementation troubles.

The first experimental phase exploited a publicly available cultural heritage dataset[1] consisting of about 8K images of 5 kinds of data (Drawings and watercolors, Works of painting, Sculpture, Graphic Art, Iconography) downloaded from the internet. For the experimental goal, only unique images of drawings, paintings, sculptures, and engravings have been retained. In particular, 51 Drawings, 142 Engravings, 222 Paintings, and 616 sculptures have been used. From each image, a set of 19 corrupted images has been derived in order to simulate actual noisy user captures from personal devices. These images were derived by using a fast and flexible image augmentation library[2]. The list of exploited transformations from the library is as follows: ShiftScaleRotate, Blur, OpticalDistortion, GridDistortion, HueSaturationValue, ISONoise, MotionBlur, RandomBrightnessContrast, RandomBrightness, RandomFog, RandomGamma, RandomShadow, RandomSunFlare, Solarize, ColorJitter, ElasticTransform, GaussianBlur, GaussNoise, GlassBlur.

Table I reports the cardinality of the images used in this first experiment. The last row reports the total number of images (overall classes): the 1031 original images of the database were retained as references whereas all the generated images (1031x19=19589) were used for testing instead. This enabled an extensive evaluation of the introduced techniques also through a quantitative comparison with classical approaches based on SIFT for localizing and describing keypoints [16].

Table II and Table III report the performance in image matching on the considered dataset. In each table the first column indicates the considered class of artworks whereas remaining columns show accuracy, precision, recall and F1-score for each class.

The second experimental phase concentrated on the specific application field of the INSTANT project. In particular, 616 images were acquired in 1) Galatina, 2) Carovigno, 3) Cavallino and 4) Patù. They are historical sites in the Salento area, south east of Italy. In each site, for each artwork (point of interest), one image was taken as a reference and the remaining ones are used for testing. Reference images were acquired by a NIKON d7200 whereas testing images were acquired by mobile phones (XIAOMI MI 9, RedMi Note 8 Pro, iPhone 8 and Samsung Note 3) belonging to different price ranges and therefore with very different technical characteristics in order to simulate real use conditions. Details about the INSTANT project image dataset are reported in Table IV. Results of matching test images with respect to reference images are reported in Table V and Table VI for SIFT and CNN approach respectively. From the tables, it can be derived that the CNN based feature extraction approach improves image matching performance,

TABLE I
The Selected Benchmark Dataset

Data Type	n. of reference images	n. of test images
Drawing	51	969
Engravings	142	2698
Paintings	222	4218
Sculptures	616	11704
Overall	**1031**	**19589**

[1] https://www.kaggle.com/thedownhill/art-images-drawings-painting-sculpture-engraving

TABLE II
RESULTS ON THE SELECTED BENCHMARK DATASET USING TRADITIONAL SIFT MATCHING

Data Type	Acc	Prec	Rec	F1-Score
Drawings	0.9772	0.9799	0.9772	0.9778
Engravings	0.9918	0.9924	0.9918	0.9918
Paintings	0.9777	0.9745	0.9777	0.9749
Sculptures	0.9658	0.9587	0.9658	0.9590
Overall	**0.9781**	**0.9763**	**0.9781**	**0.9758**

TABLE III
RESULTS ON THE SELECTED BENCHMARK DATASET USING PROPOSED CNN MODEL

Data Type	Acc	Prec	Rec	F1-Score
Drawings	0.9825	0.9900	0.9825	0.9846
Engravings	0.9956	0.9963	0.9956	0.9957
Paintings	0.9810	0.9779	0.9810	0.9782
Sculptures	0.9713	0.9616	0.9713	0.9644
Overall	**0.9826**	**0.9815**	**0.9826**	**0.9807**

especially in poorly textured object areas. The correct retrieval by CNN of the reference image of the sculpture is reported in Fig. 6. The wrong matching gathered by SIFT on the same test image can be observed in Fig. 7.

B. Performance of the INSTANT system

In order to evaluate the performance of the INSTANT system, the JMeter tool [30] has been selected. Although JMeter presents a steep learning curve, it is considered the leader of load testing tools in the open source market: it has a large active community, and the product provides a wide range of features. It supports dashboard report generation to obtain graphs and statistics from the test plan.

The performance validation highlights the results of two different types of tests: load testing and stress testing. Each test has been carried out in a virtual demo environment whose hardware and software characteristics are shown in Table VII.

1) Load Testing

Load testing aims to analyze the system's ability to respond to a predetermined load flow. In short, it refers to the practice of modeling the intended use of a software program by simulating simultaneous access by multiple users. It turns out that this kind of test is more relevant and effective for multi-user systems (such as the INSTANT system). Analysis of the results in this study was based on the results of the Summary Report produced by Apache JMeter based on the testing scenarios summarized in Table VIII.

TABLE IV
THE COMPOSITION OF THE INSTANT DATASET FRAMING THE CONSIDERED POINTS OF INTEREST IN THE SITE #1 IN GALATINA, SITE #2 IN CAROVIGNO, SITE #3 IN CAVALLINO AND SITE #4 IN PATÙ

	Reference	Test
Site #1 paintings	74	236
Site #1 sculptures	16	61
Site #1 outdoor	9	34
Total site #1	**99**	**331**
Site #2 paintings	21	47
Total site #2	**21**	**47**
Site #3 paintings	5	25
Site #3 sculptures	16	51
Total site #3	**21**	**76**
Site #4 paintings	8	31
Site #4 outdoor	5	27
Total site #4	**13**	**58**
TOTAL	**154**	**512**

TABLE V
RESULTS ON THE INSTANT DATASET BY USING SIFT APPROACH

	Acc	Prec	Rec	F1-Score
Site #1 paintings	1	1	1	1
Site #1 sculptures	0.8571	0.9166	0.9166	0.8888
Site #1 outdoor	1	1	1	1
Overall site #1	**0.9523**	**0.9722**	**0.9722**	**0.9629**
Site #2 paintings	1	1	1	1
Overall site #2	**1**	**1**	**1**	**1**
Site #3 paintings	1	1	1	1
Site #3 sculptures	0.9444	0.9583	0.9444	0.9428
Overall site #3	**0.9722**	**0.9791**	**0.9722**	**0.9714**
Site #4 outdoor	0.9130	0.9444	0.9166	0.9111
Site #4 paintings	0.9375	0.9687	0.9687	0.9642
Overall site #4	**0.9252**	**0.9565**	**0.9426**	**0.9376**
Overall	**0.9624**	**0.9769**	**0.9717**	**0.9679**

TABLE VI
RESULTS ON THE INSTANT DATASET BY USING CNN APPROACH

	Acc	Prec	Rec	F1-Score
Site #1 paintings	1	1	1	1
Site #1 sculptures	0.8571	0.9166	0.9166	0.8888
Site #1 outdoor	1	1	1	1
Overall site #1	**0.9523**	**0.9722**	**0.9722**	**0.9629**
Site #2 paintings	1	1	1	1
Overall site #2	**1**	**1**	**1**	**1**
Site #3 paintings	1	1	1	1
Site #3 sculptures	1	1	1	1
Overall site #3	**1**	**1**	**1**	**1**
Site #4 outdoor	1	1	1	1
Site #4 paintings	1	1	1	1
Overall site #4	**1**	**1**	**1**	**1**
Overall	**0.9880**	**0.9930**	**0.9930**	**0.9907**

The *Ramp-up period* is one of the main parameters used by JMeter to simulate the actual load on the web application. It represents the planned time of execution of all threads (users). Therefore, when discussing the *Ramp-up period*, it is also important to consider its relationship with threads. So, for example, if you consider 10 threads and a *Ramp-up period* of 100 seconds, it means that JMeter will start 10 threads (users) within a time frame of 100 seconds. Then, each thread will start with a delay of 10 seconds compared to the previous thread.

For each testing scenario, two different results have been reported. A *Response Time vs Threads graph* (Fig. 8) and the *APDEX (Application Performance Index) table* (Table IX). The first shows how Response Time changes with amount of parallel threads. The APDEX table, instead, reports APDEX for each transaction based on configurable values for tolerated and satisfied thresholds. In details, APDEX is an open standard used to measure the performance of software applications in computing. The standard converts many measurements into a number in a uniform scale of 0 to 1 (0 = no users satisfied, 1 = all users satisfied). To compute it JMeter needs two values: *Satisfied count* and *Tolerating count*. *Satisfied count* is the number of requests for which Response Time is lower than *Toleration threshold*. *Tolerating count* is the number of requests for which response time is higher than *Toleration threshold* but lower than *Frustration threshold*.

For space considerations, Table IX shows aggregated data obtained from each testing scenario.

Fig. 6. On the left the test image and on the right the reference image retrieved by CNN based approach. Different scale and pose did not affect the system.

Fig. 7. On the left the test image and on the right the wrong reference image retrieved by SIFT based approach. In this case the variance in pose affects the system.

TABLE VII
HARDWARE AND SOFTWARE CHARACTERISTICS OF THE VIRTUAL DEMO ENVIRONMENT

vCPU	RAM	Hard Disk	Operating System
4	4 GB	70 GB	Linux CentOS 6.8

TABLE VIII
LOADING TESTING SCENARIOS

	Number of users	Ramp-up period [s]
First scenario[a]	50	10
Second scenario[b]	200	50
Third scenario[c]	500	150
Fourth scenario[d]	1000	300
Fifth scenario[e]	2000	700

Apache JMeter takes about:
[a]10 seconds to create and run all existing threads, where each successive thread will be delayed by 200 milliseconds.
[b]50 seconds to create and run all existing threads, where each successive thread will be delayed by 250 milliseconds.
[c]150 seconds to create and run all existing threads, where each successive thread will be delayed by 300 milliseconds.
[d]300 seconds to create and run all existing threads, where each successive thread will be delayed by 300 milliseconds.
[e]700 seconds to create and run all existing threads, where each successive thread will be delayed by 350 milliseconds.

TABLE IX
LOAD TESTING: APDEX – APPLICATION PERFORMANCE INDEX

Testing scenario	APDEX	T (Toleration threshold)	F (Frustration threshold)
50 users	0.847	1 sec 500 ms	3 sec
200 users	0.716	1 sec 500 ms	3 sec
500 users	0.782	1 sec 500 ms	3 sec
1000 users	0.659	1 sec 500 ms	3 sec
2000 users	0.799	1 sec 500 ms	3 sec

TABLE X
STRESS TESTING SCENARIOS

	Number of users	Ramp-up period [s]
First scenario[a]	500	100
Second scenario[b]	1000	200
Third scenario[c]	2000	300
Fourth scenario[d]	10000	2000

Apache JMeter takes about:
[a]100 seconds to create and run all existing threads, where each successive thread will be delayed by 200 milliseconds.
[b]200 seconds to create and run all existing threads, where each successive thread will be delayed by 200 milliseconds.
[c]300 seconds to create and run all existing threads, where each successive thread will be delayed by 150 milliseconds.
[d]2000 seconds to create and run all existing threads, where each successive thread will be delayed by 200 milliseconds.

2) Stress Testing

Stress testing aims to determine or verify the performance of a system or application under conditions that exceed the level used for load testing, and then when the flow is greater than that expected in production. Unlike load testing, stress testing allows us to evaluate the robustness of the system under stress conditions and determine the limit conditions of the system to detect any abnormal system behavior and how end users perceive these behaviors. In the load testing the reported error rate for simultaneous access by 2000 users was 1.07%, and the total APDEX index was 0.799, which exceeded the tolerance threshold. Starting from these results, the threshold of 2000 users has been exceeded in the stress testing, gradually impairing the performance of the system. Four testing scenarios have been carried out by considering 500 users, 1000 users, 2000 users, and 10000 users, respectively, but setting a lower Ramp-up period.

Four test scenarios have been configured in JMeter with the settings summarized in Table X.

The APDEX indexes obtained in stress testing are reported in Table XI. Also in this case, the aggregate data obtained from each test plan have been reported.

The obtained Time vs Threads graphs have been reported in Fig. 9. It can be seen that, after reaching a thousand users (at a lower Rump-up period), the system performance began to decline, and the high error rate following the calls at POI services damaged a large part of the overall platform availability. With a Rump-up period of 300 seconds for 2000 users, the reported error rate has increased by almost 20%, and many functions became usable within an acceptable time.

Based on the results obtained in previous tests, assumptions about the user traffic tolerated by the system can be expressed. The load testing showed a very low error rate and excellent response time up to 2000 user visits in 700 seconds. When the user traffic exceeds 300 simultaneous visits, the load testing and

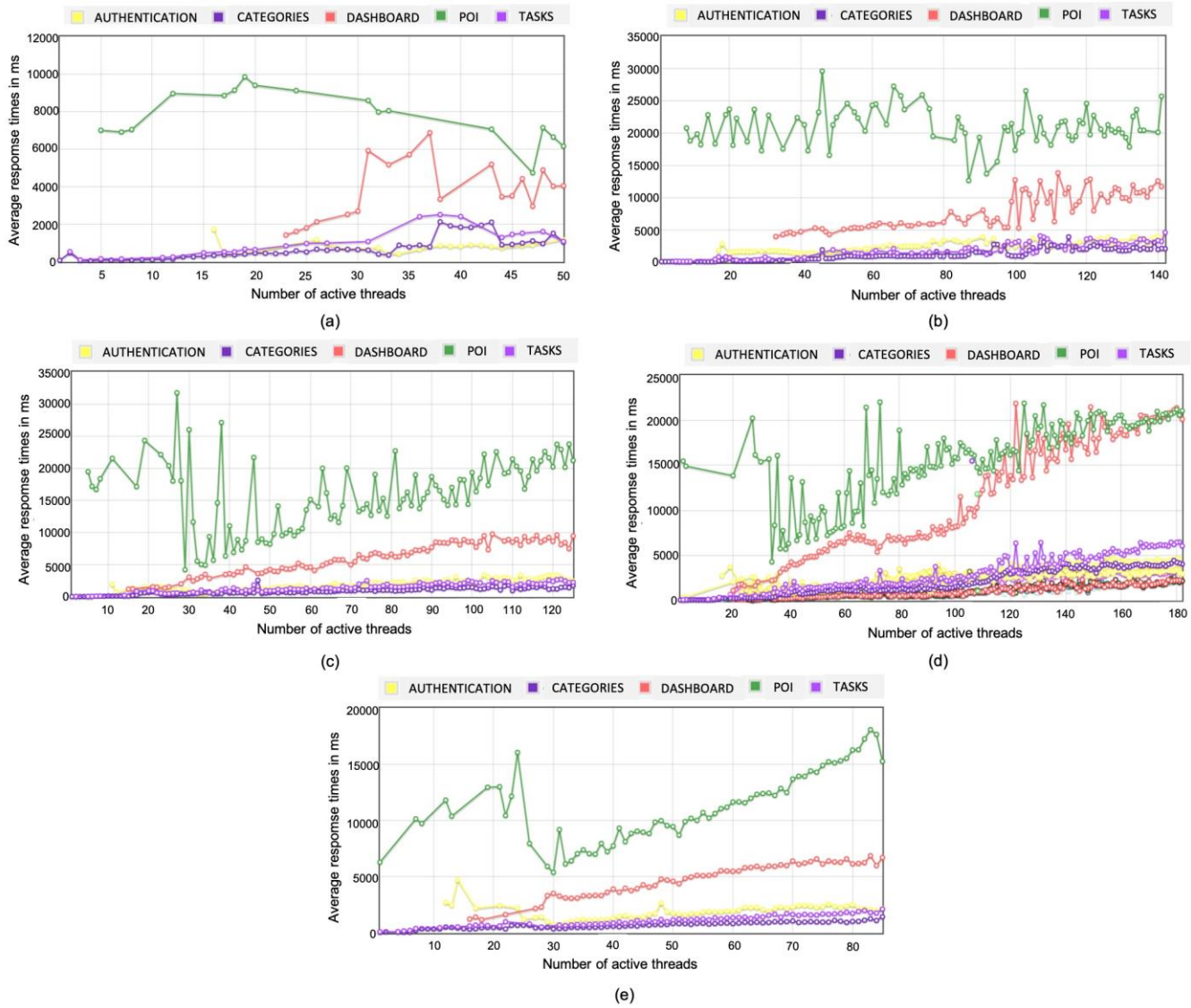

Fig. 8. Time vs Threads graphs obtained in load testing. First scenario, 50 users in 10 seconds (a); second scenario, 200 users in 50 seconds (b); third scenario, 500 users in 150 seconds (c); fourth scenario, 1000 users in 300 seconds (d); fifth scenario, 2000 users in 700 seconds (e).

TABLE XI
STRESS TESTING: APDEX – APPLICATION PERFORMANCE INDEX

Testing scenario	APDEX	T (Toleration threshold)	F (Frustration threshold)
50 0users	0.517	1 sec 500 ms	3 sec
1000 users	0.335	1 sec 500 ms	3 sec
2000 users	0.289	1 sec 500 ms	3 sec
10000 users	0.280	1 sec 500 ms	3 sec

stress testing show obvious exponential sensitivity. In order to more accurately determine the ideal user flow, variables based on user behavior should also be considered, such as bounce rate, total session duration, and consultation of previously used services. If these variables are also considered, the system may show better performance. Instead, the tests carried out simulate a predetermined path with a predetermined time and invokes all the services offered by the INSTANT system. However, it can be concluded that approximately 2000 users can utilize the full potential of the system in 700 seconds without encountering errors. In addition, using the current microservices architecture, the system administrator can monitor performance changes in real time, and when problems arise, s/he can rely on many configurations, including increased computing power. Therefore, scalability is one of the main features of the proposed system, which can be used to dynamically respond to potentially larger user flows.

VII. CONCLUSION

This paper proposes an innovative system based on IoT technologies and deep learning methods, which aims to enhance art and cultural heritage and promote a personalized user experience when visiting art and cultural places. The developed architecture is based on a modern microservices approach, which results in a highly scalable and flexible platform. Each service has been built, deployed and is scalable separately and communicates with other services through standardized APIs. The validation of the computer vision module proved the high performance of the approach adopted, and the functional and

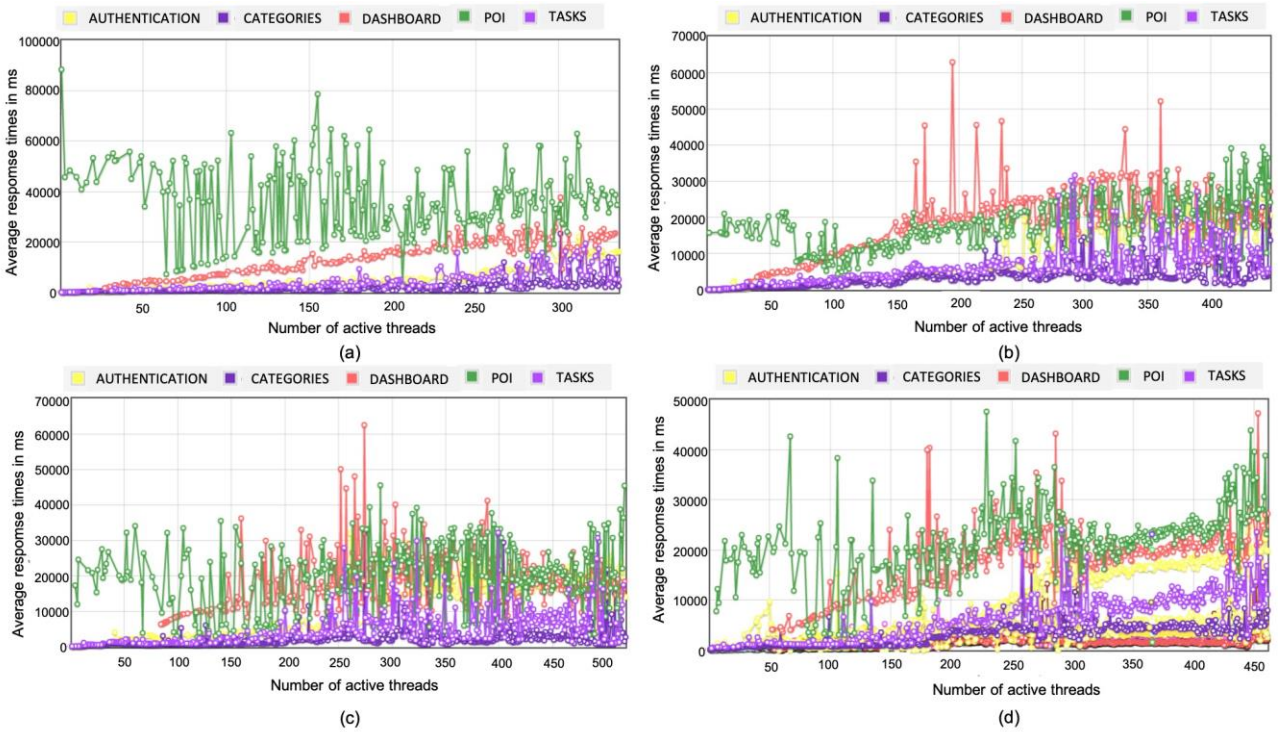

Fig. 9. Time vs Threads graphs obtained in stress testing. First scenario, 500 users in 100 seconds (a); second scenario, 1000 users in 200 seconds (b); third scenario, 2000 users in 300 seconds (c); fourth scenario, 10000 users in 2000 seconds (d).

performance validation of the entire system proved its effectiveness and its ability to maintain an important user flow.

Future works will deal with the development of further personalized services for users to suggest trips, events and other travel experiences based on their interests and needs. Event providers will be able to use the platform to insert events (cooking classes, music events, food and wine tours, etc.) and related details. Tourists will be able to enter some simple information related to their needs and interests via the mobile application (dates, children's presence, interest in food and wine events or music events, etc.). Therefore, the platform will utilize algorithms based on machine learning, which will provide tailored results for users based on their profile, needs, interests, and their previous interactions with the mobile application.

REFERENCES

[1] S. Nižetić, P. Šolić, D. López-de-Ipiña González-de-Artaza, L. Patrono, "Internet of things (iot): Opportunities, issues and challenges towards a smart and sustainable future," *J. Clean. Prod.*, vol. 274, Nov. 2020. DOI: 110.1016/j.jclepro.2020.122877.

[2] J. Díaz-de-Arcaya, R. Miñón, A. I. Torre-Bastida, J. Del Ser, A. Almeida, "PADL: A Modeling and Deployment Language for Advanced Analytical Services", *Sensors*, vol. 20, n. 23, 6712, 2020. DOI: 10.3390/s20236712

[3] L. Patrono, L. Atzori, P. Šolić, M. Mongiello, A. Almeida, "Challenges to be addressed to realize Internet of Things solutions for smart environments," Future Gener. Comput. Syst, vol. 111, pp.873-878, 2020. DOI:10.1016/j.future.2019.09.033.

[4] Z. Luo, L. Zhou, X. Bai, H. Chen, J. Zhang, Y. Yao, S. Li, T. Fang, L. Quan, "ASLfeat: Learning local features of accurate shape and localization," *in Proc. CVPR 2020*, pp. 6588–6597. DOI: 10.1109/CVPR42600.2020.00662.

[5] F. Barile, D. Calandra, A. Caso, D. D'Auria, D. Di Mauro, F. Cutugno, S. Rossi, "ICT solutions for the or.c.he.s.t.r.a. project: From personalized selection to enhanced fruition of cultural heritage data," *in Proc. SITIS 2014*, Marrakech, Morocco, 23-27 Nov. 2014. DOI: 10.1109/SITIS.2014.12.

[6] A. Perles, E. Perez-Marin, R. Mercado, J. Segrelles, I. Blanquer, M. Zarzo, F. Garcia-Diego, "An energy-efficient Internet of Things (IoT) architecture for preventive conservation of cultural heritage," *Future Gener. Comput. Syst.*, vol. 81, pp. 566–581, 2018. DOI: 10.1016/j.future.2017.06.030.

[7] I.R. Polyakova, G. Maglieri, S. Mirri, P. Salomoni, R. Mazzeo, "Art scene investigation: discovering and supporting cultural heritage conservation through mobile AR," *in Proc. IEEE INFOCOM 2019*, Paris, France, 29 Apr. - 2 May 2019. DOI: 10.1109/INFCOMW.2019.8845310.

[8] S. Alletto, R. Cucchiara, G. Del Fiore, L. Mainetti, V. Mighali, L. Patrono, G., "An indoor location-aware system for an iot-based smart museum," *IEEE Internet Things J.*, vol. 3, pp. 244–253, 2020. DOI: 10.1109/JIOT.2015.2506258.

[9] V. Mighali, G. Del Fiore, L. Patrono, L. Mainetti, S. Alletto, G. Serra, R. Cucchiara, "Innovative iot-aware services for a smart museum," *in Proc. WWW '15*, Florence Italy, 18 – 22 May 2015, pp. 547–550. DOI: 10.1145/2740908.2744711.

[10] A. Belhi, A. Bouras, "CNN features vs classical features for large scale cultural image retrieval," *in Proc. ICIoT 2020*, Doha, Qatar, 2-5 Feb. 2020. DOI: 10.1109/ICIoT48696.2020.9089643.

[11] M. La Franca, L. Marino, L. Martorana, M. Leo, P. Carcagnì, C. Distante, I. Sergi, L. Patrono, "A novel approach based on microservices architectures and computer vision to improve access to culture heritage," *in Proc. SpliTech 2020*, Split, Croatia, 23-26 Sept. 2020. DOI: 10.23919/SpliTech49282.2020.9243806.

[12] A. Saini, T. Gupta, R. Kumar, A. Gupta, M. Panwar, A. Mittal, "Image based Indian monument recognition using convoluted neural networks," *in Proc. BID 2017*, Pune, India, 20-22 Dec. 2017. DOI: 10.1109/BID.2017.8336587.

[13] M. Fiorucci, M. Khoroshiltseva, M. Pontil, A. Traviglia, A. Del Bue, S. James, "Machine learning for cultural heritage: A survey," *Pattern Recognit. Lett.*, vol. 133, pp. 102-108, 2020. DOI: 10.1016/j.patrec.2020.02.017.

[14] W. Yu, X. Sun, K. Yang, Y. Rui, H. Yao, "Hierarchical semantic image matching using CNN feature pyramid," Comput. Vis. Image Underst., vol. 169, pp. 40–51, 2018. DOI: 10.1016/j.cviu.2018.01.001.

[15] C. Leng, H. Zhang, B. Li, G. Cai, Z. Pei, L. He," Local feature descriptor for image matching: A survey," *IEEE Access*, vol. 7, pp.6424–6434, 2018. DOI: 10.1109/ACCESS.2018.2888856.

[16] D.G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, n. 2, pp. 91–110, 2004. DOI: 10.1023/B:VISI.0000029664.99615.94.
[17] M. Dusmanu, I. Rocco, T. Pajdla, M. Pollefeys, J. Sivic, A. Torii, T. Sattler, "D2-Net: A Trainable CNN for Joint Detection and Description of Local Features," *in Proc. CVPR 2019*, Long Beach, CA, USA, 15-20 June 2019, pp. 8092-8101. DOI: 10.1109/CVPR.2019.00828.
[18] Y. Ono, E. Trulls, P. Fua, K.M. Yi, "Lf-Net: Learning local features from images," *in Proc. NeurIPS 2018*, Montrèal, Canada, 3-8 Dec. 2018, pp. 6237–6247. DOI: 10.5555/3327345.3327521.
[19] J. Ma, X. Jiang, A. Fan, J. Jiang, J. Yan, "Image matching from handcrafted to deep features: A survey," *Int. J. Comput. Vis.*, vol. 129, pp. 23–79, 2020. DOI: 10.1007/S11263-020-01359-2.
[20] Drifty Co., Ionic Framework. Official website. [Online]. Available: https://ionicframework.com/. (accessed: 20.05.2021).
[21] M. Otto and J. Thornton, Bootstrap framework. Official website. [Online] Available: https://getbootstrap.com/. (accessed: 23.04.2021).
[22] Google LLC, Angular platform. Official website. [Online] Available: https://angular.io/. (accessed: 23.04.2021).
[23] IETF, WebSocket protocol. RFC 6455. [Online]. Available: https://tools.ietf.org/html/rfc6455. (accessed: 23.04.2021).
[24] P. Precht, J. Philipp, M. Prichinenko, D. Valero, B. Longaeret, et al., Angular translate module. [Online] Available: https://angular-translate.github.io/. (accessed: 23.04.2021).
[25] IETF, JSON Web Token (JWT). RFC 7519. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc7519. (accessed: 23.04.2021).
[26] SmartBear Software, Swagger. Official website. [Online]. Available: https://swagger.io/. (accessed:23.04.2021).
[27] Oracle Corporation, MySQL database. Official website. [Online]. Available: https://www.mysql.com/. (accessed: 23.04.2021).
[28] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, Y. Wei, "Deformable convolutional networks," *in Proc ICCV 2020*, pp. 764–773, 2017. DOI: 10.1109/ICCV.2017.89.
[29] D. Mishkin, F. Radenovic, J. Matas, J., "Repeatability is not enough: Learning affine regions via discriminability," *in Proc. ECCV 2018*, Munich, Germany, 8-14 Sept. 2018, pp.287–304. DOI: 10.1007/978-3-030-01240-3_18.
[30] Apache Software Foundation, JMeter. [Online] Available: https://jmeter.apache.org/. (accessed: 23.04.2021).
[31] H. Bay, T. Tuytelaars, L. V. Gool, "Surf: Speeded up robust features". Computer Vision and Image Understanding (CVIU), Vol. 110, No. 3, pp. 346–359, 2008.
[32] S. A. K. Tareen, Z. Saleem. "A comparative analysis of sift, surf, kaze, akaze, orb, and brisk." 2018 International conference on computing, mathematics and engineering technologies (iCoMET). IEEE, 2018.

Ilaria Sergi received the master's degree in Automation Engineering from the University of Salento, Lecce, Italy, in 2012. Her thesis focused on the tracking of small laboratory animals, based on passive UHF RFID technology. Since 2012, she has been collaborating with the Identification Automation Laboratory (IDA Lab) of the Department of Engineering for Innovation, University of Salento. In 2019 she received the PhD in Engineering of Complex Systems from the University of Salento, Italy. Her research interests include RFID, Bluetooth, Internet of Things, smart environments, and homecare solutions. She has authored several papers on international journals and conferences.

Marco Leo received an Honours Laurea Degree in Computer Engineering from the University of Salento (Italy) in 2001. Since then, he has been working as a Researcher at the CNR, Italy. His main research interests include computer vision and pattern recognition. He participated in a number of national and international research projects focusing on assistive technologies, automatic video surveillance, human attention monitoring, real-time event, and non-destructive inspection of aircraft components. He is author of more than 100 papers published in international journals and conferences. He is also a co-author of three international patents on visual systems for event detection in sport contexts.

Pierluigi Carcagnì received the degree in Computer Engineering in 2002 from the University of Salento. From 2003 to 2015 he worked at CNR-INO in development of optoelectronic instrumentation, IR and color high definition reflectography systems, multispectral and hyperspectral analysis systems, laser triangulation systems, color measurement systems. He was a founder of the CNR's spin-off Company Taggalo. Since 2015 he has been working at CNR-ISASI where his main research interests include computer vision and pattern recognition. In 2016 he received the PhD in Computer Engineering from the University of Salento with topics related to computer vision technique.

Marco La Franca received the degree in Computer Science in 2006 at the University of Palermo. From 2006 to 2008 he worked in Palermo for Datel Technology (later Sempla) as Developer, Analyst and Team Leader. From 2009 to 2012 he moved to Turin where he worked as Analyst and Team Leader for Sempla (later GFT), dealing with important projects in the banking sector (Intesa-SanPaolo). Since 2012 he has returned to Palermo and worked in Arancia-ICT as software architect, team leader, project manager and partner. He managed innovative digital transformation projects, focused on Artificial Intelligence, Image and Data Analysis and Blockchain. Since 2021 he works in Cloudtec s.r.l. as Platforms Delivery Manager.

Cosimo Distante received the master's in Computer Science at the University of Bari, and PhD in Engineering at the University of Salento (Italy). His main expertise is in Computer Vision and Pattern Recognition, Image Processing, Machine Learning and Robotics. He joined in 2001 the CNR, Italy. Since 2003 he has been working as Contract Professor for Computer Vision, Pattern Recognition and Image Processing at the University of Salento. He founded the Taggalo CNR's spinoff company. He is unit manager of the Institute of Applied Sciences and Intelligent System of the CNR. His main interests include medical imaging, video surveillance and machine vision for inspection.

Luigi Patrono received the M.S. degree in Computer Engineering from the University of Lecce, Italy, in 1999, and the Ph.D. degree in innovative materials and technologies for satellite networks from the ISUFI-University of Lecce, in 2003. He is Associate Professor of Computer Networks and Internet of Things and also the Pro-Vice Chancellor for Digital Technologies at the University of Salento. His research interests include RFID, IoT, WSN, and embedded systems. He has authored more than 140 scientific papers published in international journals and conferences. He is the Organizing Chair of some international symposia and workshops focused on the IoT.