CHAPTER

Wearable Systems for Improving Tourist Experience

1

Lorenzo Seidenari[1,a], Claudio Baecchi[1], Tiberio Uricchio, Andrea Ferracani, Marco Bertini and Alberto Del Bimbo

[*] *University of Florence, Viale Morgagni, 65 Firenze*

[a] *Corresponding:* lorenzo.seidenari@unifi.it

CHAPTER OUTLINE HEAD

ABSTRACT

In this chapter we present original approaches for the development of a smart audio guide that adapts to the actions and interests of visitors of cultural heritage sites and exhibitions either in indoor or outdoor scenarios. The guide is capable of perceiving the context. It understands what the user is looking at, if he is moving or is inattentive (e.g. talking with someone), in order to provide relevant information at the appropriate timing. Automatic recognition of artworks is performed with different approaches depending on the scenario, i.e. indoor and outdoor. These approaches are respectively based on Convolutional Neural Network (CNN) and SIFT descriptors, performing, when appropriated, object localization and classification. The computer vision system works in real-time on the mobile device, exploiting also a fusion of audio

[1] Equal contribution.

and motion sensors. Configurable interfaces to ease interaction and fruition of multimedia insights are provided for both scenarios. The audio-guide has been deployed on a NVIDIA Jetson TX1 and a NVIDIA Shield Tablet K1, tested in a real world environment (Bargello Museum of Florence and the historical city center of Florence), and evaluated with regard to system usability.

Keywords:
Deep Learning, Computer Vision, Object Detection, Image Retrieval, Mobile Computing, Cultural Heritage, Audio Guide, SIFT

1.1 INTRODUCTION

Natural and adaptive interfaces exploiting recent progresses in mobile, multimedia and computer vision technologies can effectively improve user experience during a a visit of a cultural heritage location or a museum. Multimodal data coming from mobile and wearable device sensors can be processed, analysed and fused onboard in real-time in order to automatically understand the context of the visit, user's needs, behaviour and interests, e.g. what the visitor is looking at, for how long, and if other events happen during the visit itself.

In particular, visual object classification can help in disambiguating where user's attention is directed by observing what he is looking at, as well as understanding if other people are occluding his view. Data from localization and motion sensors, such as GPS, gyroscope and accelerometer, can give insights about the user position with respect to Point-Of-Interests (POIs). His movement in direction and velocity can help to understand if the visitor is just wandering through the museum, or if he is looking at an exhibit that really interests him. Finally, voice activity detection (VAD) can reveal, if the visitor has friends that accompany him during the visit, if he is busy in a conversation. These results are particularly suited to the intended use of smart audio guides that should accompany and help the user, as a real museum guide, to satisfy curiosities and respond to personal needs [6].

However, approaches for inferring the context and detecting user behaviour exploiting device sensors must necessarily be different in case of indoor or outdoor scenarios, given the extreme diversity of these situations. In the outdoor scenario, GPS provides a fairly reliable localization which is hardly available indoor. Conversely, audio sensing is extremely limited outdoor, due to the poor quality of the acquired signal. Considering these problems, in the following we present two versions of the audio guide, one designed for the indoor scenario and the other for the outdoor. These two versions share the same human-computer interface for a fully automatic use: the system is worn and the output is provided through speech synthesis and present slight difference in case some user interaction is needed.

1.2 RELATED WORK

Personalized museum experience.

Recent statistics from the US National Travel and Tourism Office reveal a new record of tourism-related activities[1]. Despite their simplicity, audio guides are still one of the most used interactive devices in a museum visit [5]. In recent years there has been a growing interest in improving user experience using smart digital guides. In fact, although classic human guides are still an available option, the modern tourist, expects new personal augmented experiences, founded on individual preferences, context awareness and ease of use, possibly using their personal device.

Many works in the literature propose the use of mobile systems to have an augmented personalized experience in the context of the visit to cultural heritage sites. The need of personalization and user adaptation has recently become of great interest in research [6]; a review of the different types of smart mobile guides addressing the problem of personalization has been provided in [3]. The work of Abowd *et al.* [1] is one of the first on this subject, marking the difference between systems which provide improved experiences in indoor (e.g. museums) and outdoor scenarios (e.g. monuments visible traveling in a city). This difference, from a system implementation perspective, is mainly that in indoor there is more control on the dataset of artworks to be evaluated with respect to the outdoor scenario where this control is missing. The distinction, as detailed later in this Chapter, is especially important in the case where the system must scale in performing non-trivial tasks hosted onboard, such object classification and artwork recognition, tasks that may rely on the availability of an adequate training dataset.

Using sensors and cameras it is possible to localize visitors in proximity of artworks, thus inferring an implicit interest. In [44] the Cultural Heritage Information Personalization (CHIP) system has been proposed, where a personalised visit could be created through a web interface. The tour can be downloaded to a mobile device. Track of the artworks visited by the user in the museum is obtained using RFIDs. Information is then synced back on the server side in order to update the user profile and provide a better personalisation for next tours. In [48] and [26] the behavioural patterns of museum visitors is analysed and predicted according to four main patterns that emerged from ethnographic observations by [13]. The works show that these patterns can be identified exploiting features such as average time spent on each artwork, percentage of observed artworks, etc. [21] presents a non-intrusive computer-vision system based on person re-identification of museum visitors observed through surveillance cameras. The system tracks visitors behaviour in the museum and measures the time spent in looking at each artwork. In this way a personalised user profile of interests is built. The user profile is then used at the end of the visit to adapt an interface and recommend targeted multimedia content on a wide interactive table located at the exit of the exhibit. In this way the system can pro-

[1] `http://tinet.ita.doc.gov/tinews/archive/tinews2017/20170413.asp`

vide more information on the artworks that attracted the visitor the most, and suggest additional targeted tours in outdoor.

Regarding outdoor systems and the more broad context of city tourism, several notable works propose context aware mobile guides that can provide information depending on the user position and contextual cues [10, 15, 22] such as time, user profile, device profile or network conditions. In [38], one of the first software for a tourist guide was presented. It shows areas of interest on a map as well as available public utility points, by also exploiting the GPS location of the user. Takeuchi and Sugimoto developed the CityVoyager system [39] that uses GPS to detect user location, understand visited places such as shops and suggest new items based on user preferences. To reduce the cognitive effort needed to use the system, it features a simple "metal detector" interface that beeps faster when recommended shops are nearby. In [24], a mobile agent that automatically selects attractions, plans individual tours following tourist's specific behaviour was proposed. The focus is on planning and group behaviour analysis. A wearable device, sharing information with a handheld device, was used in [23] to develop an un-intrusive mobile tour guide with real time updates.

Furthermore, thanks to the recent developments in mobile computing, some advanced solutions for mobile guides with embedded processing capabilities have been proposed. Augmented Reality solutions have been exploited in specific domains. For instance, in [19], a computer vision based mobile guide, able to automatically recognize and enrich artworks by overlaying information, was developed for religious museums. However, this class of solutions are distracting and highly cumbersome since they require the user to constantly interact with the device, interrupting the experiential flow. RFIDs have been applied to tags placed near artworks and scanned by mobile phones [8, 43]. Nonetheless, this approach is still limited to controlled environments, it scales poorly and is unsuitable for outdoor applications. Wearable systems have been used to interact with artworks [4]. This approach is the less intrusive and more immersive, since it adopts a first person perspective, in the sense that that a wearable system equipped with a camera and pointed in the field of view can process images exactly as the user see the scene.

Object detection and recognition

Recent object detection methods are all based on modern deep convolutional neural networks [25]. To avoid the computation of a CNN forward pass on multiple subwindows [18], recent approaches apply the object classification and bounding box regression stage directly on the last convolutional feature map of the CNN [17, 36]. An even faster strategy is to directly generate a set of class-labeled bounding boxes with a single pass of a convolutional network. Redmon *et al.* [35] argue that "You should Only Look Once" (YOLO) at frames. This idea treats the task of object detection as a regression problem, generating location and scores directly. Liu *et al.* [28] proposed an approach named Single-Shot Detection (SSD), which is very similar to YOLO with the difference that multiple convolutional maps are used in

order to evaluate more windows, at multiple scales.

Availability of multi-core CPUs and GPUs on mobile devices has recently allowed to implement CNNs on smartphones. In [47] an analysis of the best CNN architectures for mobile devices has been performed, evaluating the effect of BLAS routines and NEON SIMD instructions of the ARM CPUs. The use of weight quantization is employed to deal with reduced memory capability [30, 40, 46]. In [20] a framework to execute deep learning algorithms on mobile devices has been presented, using OpenCL to exploit the GPUs. In [27] has been presented an open source framework for GPU-accelerated CNNs on Android devices, that parallelizes more computationally intensive types of layers on GPUs and the execution of the others on CPUs using SIMD instructions.

Content-based retrieval for Cultural Heritage

Over the years different methods and applications of content-based image retrieval (CBIR) techniques have been proposed for the cultural heritage domain. Retrieving 3D artworks, e.g. statues, using salient SIFTs has been proposed in [11]; mutual information is used to filter background features. Fisher Vectors are employed to recognize style and author in [2]. In [34] has been presented a comparison of different techniques for image classification and retrieval in cultural heritage archives, using engineered and learned features. This work highlights two issues when applying current state-of-the-art CBIR techniques for cultural heritage: *i)* the need to account both for micro properties, e.g. brush strokes, and macro properties, e.g. scene layout, in the design of similarity metrics; *ii)* datasets that are relatively small, with few images for each item, thus hampering methods requiring large scale training datasets.

Voice Activity Detection

Voice activity detection (VAD) is the process of detecting human speech in a audio stream; this task is essential to improve further processing like automatic speech recognition or audio compression.

Early approaches to this problem were based on heuristics and simple energy modeling, by thresholding or observing zero-crossing rate rules [45]. These methods work well when no background noise is present. Successive methods have addressed this limitation using autoregressive models and line spectral frequencies [32] to observe signal statistics in current frame and comparing it with the estimated noise statistics. Most of these algorithms assume that noise statistics are stationary over long periods of time. Therefore, given the extreme diversity and rapid changes of noise in different environments, they can not detect occasional presence of speech. More recent data-driven approaches avoid to make assumption over the noise distribution. They usually use a classifier trained to predict speech vs. non-speech given some acoustic features [14, 31]. However, their performance degrades when the background noise resembles that of speech. Current state-of-the-art methods exploit long-span context features learned through the use of recurrent neural networks [12, 16, 42] to adapt the classification on the basis of the previous frames.

1.3 BEHAVIOR ANALYSIS FOR SMART GUIDES

Audioguides come into play in different scenarios. In our analysis there is one key difference that can be made, also leading to slightly different choices in sensor integration and recognition algorithms. We differentiate between indoor and outdoor scenarios. In both cases we aim to provide information meeting the user will of knowledge and predicting correctly user attention towards elements of the environment. The only sensor that can be used reliably in both scenarios is the camera.

User location can not be easily inferred indoor, unless a WiFi positioning system is in place. On the other hand guides to be used indoor have usually a fixed set of objects to be recognized and content can be provided in advance. In the outdoor scenario it is hard to pre-load content related to all possible landmarks a user may encounter. Nonetheless GPS, and A-GPS, is often reliable enough to provide an approximate location of the user limiting the amount of objects of interest.

Recognition of objects of interests can be obtained with a simple object detection/recognition framework, using a CNN trained for artwork detection and the very same features for recognition. This framework assumes objects to be *detectable* in frames, meaning they have a well defined contour so that we can unambiguously understand user attention.

Outdoor landmark recognition faces a few more challenges with respect to indoor artwork recognition. First of all landmark have a more diverse distribution, including building, statues and archaeological sites, to name a few. Linked to this diverse set of data are different fruition strategies. While looking at a statue in a museum or in a public square follow similar behavioral patterns, visiting a vast ruins site or observing a large cathedral give rise to imagery which is much far apart. Buildings have a well defined physical limit but they are hardly visible as a whole during a visit. In case sites are to be considered, global views are never the case and visitors often stroll around an area which is pretty distinguishable. In both this emblematic outdoor use cases, unfortunately a framework based on object detection and recognition is hard to deploy. For the outdoor use case we advocate the use of local features. This paradigm allows matching partial views of objects and perform geometric verification. Interestingly the matching database can be built online exploiting even noisy geo-localizations.

1.4 THE INDOOR SYSTEM

As mentioned before, the indoor context has different constraints with respect to the outdoor one. In this case, several components can be used jointly to understand the context. Fig. 1.1 shows the architectural diagram illustrating the main submodules of the system.

The system may be divided into two main parts that are responsible for the pri-

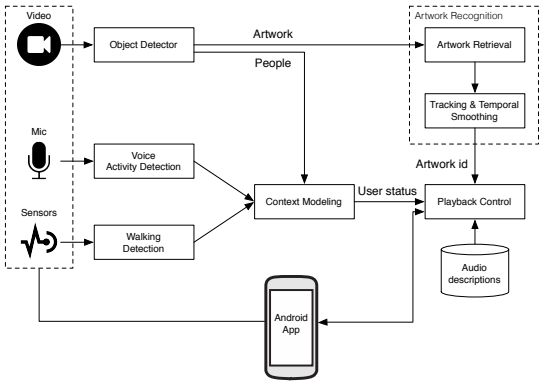

Figure 1.1: The overall system architecture.

mary functions of the system. The first one is responsible to model the *User status* while the other one to recognize artworks, i.e. provide the *Artwork id*. Together they work with the *Playback Control* component by providing input signals that enable the system to play descriptions at appropriate time.

Inputs to the systems are the camera, the microphone and the movement sensors, accessed through an Android App that also operate as front-end of the system. A computer vision system look at the scene through the camera, detects objects (*Object Detector*) and recognizes what artwork the user is looking at (*Artwork Recognition*). Artwork recognition is performed with a two-step process: first a database of known artworks is used to retrieve the most similar artwork, then tracking is performed to filter out wrong predictions.

In order to generate the *User Status* signal, the *Context Modeling* module exploits the three inputs to generate behavioral signals. The microphone as a source to the *Voice Activity Detection* module, the movement sensors to perform Walking Detection and the camera to detect *People*. They are all used to understand if the user is actively looking at an artwork or if he is engaging other activities.

Artwork Detection and Recognition

Artwork recognition is performed with a computer vision system that simultaneously does artwork localization and recognition. First the system detects the two relevant object categories for our task, i.e. persons and artworks. Then, for every detected artwork, the system perform recognition of the specific artwork. Considering that the system has a sequence of frames in input, we also take advantage from the temporal coherence to filter out spurious detections.

We base our object detector on YOLO [35], using its *Tiny Net* version, fine-tuned on our artworks dataset. Using a small CNN is mandatory for real-time requirements of our mobile platform. We train it to detect *people* and *artworks*. People detection may be used to improve context understanding and it is also fundamental to avoid

false positive detections on people, since artworks present in our training set are often statues depicting of humans.

The detection is performed as follow. For each frame, we generate scored boxes for the two required categories as follow. After splitting the original image in 7×7 blocks, each of the 49 regions (which have a size of $5 \times 2 \times |C|$) encodes the box predictions as a tuple $\langle x, y, h, w, s \rangle$. Then, maximal suppression is used to remove multiple predictions for the same objects. The confidence s represent the probability of that class being present inside the bounding box.

Artwork recognition. After detecting an artwork, the system performs recognition by re-using the computed activations of convolutional layers in the detection phase as features. A low dimensional descriptor of a region is obtained by applying a global max-pooling over feature activation maps of two convolutional layers. We selected features from layers 3 and 4 by experimental evaluation, as detailed in Sect.1.4.

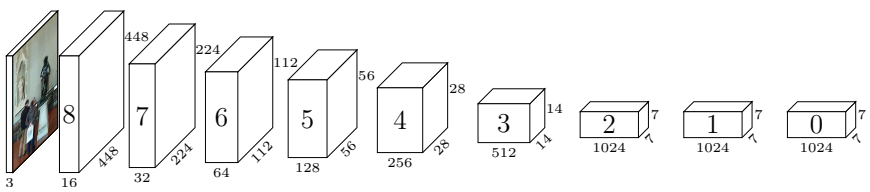

Figure 1.2: Our network architecture, with tensor size and layer numbering.

Recognition is performed with a simple 1-NN classifier. Using non-parametric classifiers allows the artwork dataset to grow without the need of re-training.

Even in case of very few wrong recognitions, user experience would be compromised since the audio guide would present different information with respect to the observed artwork. To avoid this we use three temporal processing strategies: *Distance*: we discard recognitions from artwork with a small pixel area, that may be erroneous (see Fig. 1.3). *Consistency*: we consider correct a recognition only after a certain amount of consistent labels (see Fig. 1.4). *Persistence*: it is unlikely that the user moves quickly from an artwork to another in just few frames. So, after the system recognizes an artwork, it continuously output its label proportionally to the elapsed time since the recognition.

Context Modeling

Embedded devices can be easily packed with different sensors. A smart guide should act as an autonomous agent automatically understanding the right time to engage users and when to avoid audio feedback. Independently from the acquired visual information we must understand if the user behavior is changing, e.g. due to interaction

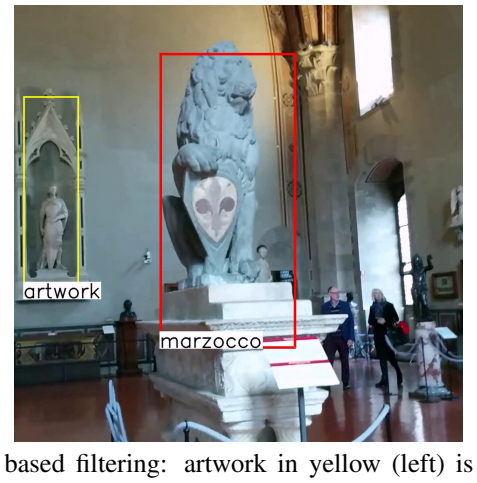

Figure 1.3: Shape based filtering: artwork in yellow (left) is not considered for recognition (too small) while the other is recognized as "marzocco" (the heraldic lion symbol of Florence).

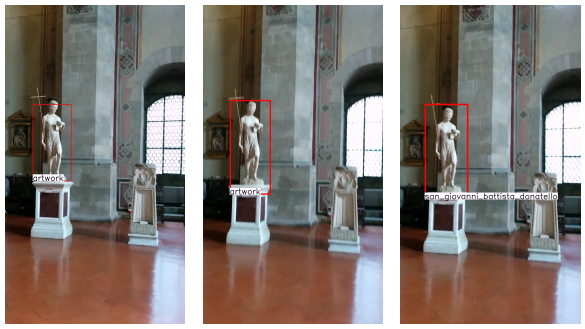

Figure 1.4: Example of artwork tracking. Only after a stable recognition over M frames ($M = 15$ in this example) the system labels the artwork.

with another person.

Detecting conversations. We want audio feedback to pause in case a conversation is happening. Hearing multiple superposed speeches is certainly a cause of disturbance, not delivering the right information and, moreover, degrading the user experience. This mechanism should come into play even if the user is standing in front of an artwork, thus regardless of the visual feedback. In the indoor scenario, we assume a high signal to noise ratio and exploit the device microphone to detect nearby speech. Although museums are typically quiet environments, in some cases there can be background music or some other environmental noise. We use a Voice Activity Detection (VAD) System, based on the implementation and model avail-

able in the OpenSMILE framework [2]. The method is based on LSTM modelling long-term signal relationship and it is able to model environmental noise [16].

Once another voice is detected, the audio playback will stop, therefore in case of false positive the user experience will be compromised. To improve the classifier quality we process an entire second of audio sampling every 0.01 seconds and averaging all predictions for the final conversation detection score.

Sensors for walking detection. User movements are an extremely important hint for context understanding. The act of standing still, walking or staying sit give important cues to interpret the attention of an user towards an artwork. In fact:

- A fast walking behavior can be safely interpreted as a sign of no interest in the visible artworks, therefore we can avoid giving feedback in such situations.
- In case the user is standing still and is listening to a description, we must avoid interrupting the audio, even in case visual cues get lost, e.g. due to temporary occlusions.

We detect walking using accelerometer data. Specifically we estimate mean and standard deviation and, after subtracting the means, we detect peaks above the standard deviation considering them as steps. If at least a step is made in the last 1 second window, we consider the person walking.

Orientation change can be detected using gyroscope data. After averaging the orientation vector on the same 1 second window we consider the facing direction changed if differing for more than 45°.

User Interface. SeeForMe has been designed with the aim to handle three different scenarios: *(i)* the user puts the device in a front pocket with the camera facing forward in order to use the system in a fully automated mode. Recognition is performed by the system on the camera stream. The user can start/stop audio reproduction through voice commands. *(ii)* the user exploits the application in a semi-automated way pointing the camera to the artworks of interest; *(iii)* the user revisits his experience once the visit is completed through a visual history of the tour in the form of a carousel.

In Fig. 1.5 two use cases of the application interface in semi-automatic mode are shown. In Fig. 1.5 *left)* The GUI presents recognized artworks through overlaid green icons that provide information on the artwork and the possibility to start/pause audio playback. If the visitor gets away from the artwork the audio played will fade out, avoiding an abrupt interruption. In case the visitor approaches back the same artwork the audio feedback is resumed automatically from where it stopped.

In Fig. 1.5 *right)* the app is used in fully automatic mode: the user is speaking with a person. The app understands the situation detecting the voices through VAD and consequently stops audio playback. Once the conversation is finished and the

[2] http://audeering.com/research/opensmile/

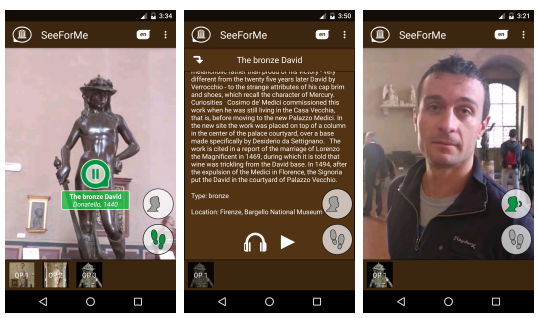

Figure 1.5: *(left)* the user is listening to the description of the artwork, *(center)* the user is reviewing an item in the history, *(right)* the user is speaking with someone not focusing on any artwork.

attention of the visitor turns to the artwork, the system automatically resumes the playback.

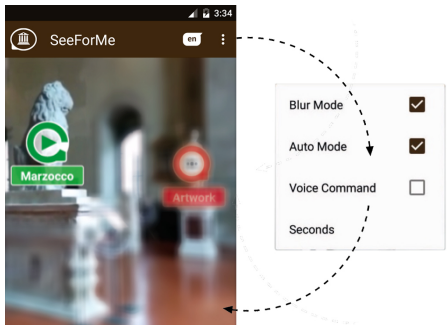

Figure 1.6: The contextual menus to configure the app properties. Here it is shown the appearance of the interface in blur mode.

Some configuration properties are available to users to manage the user interface as shown in Fig. 1.6.

The mobile app has been developed using the Android SDK. The interface follows the guidelines of Material Design proposed by Google. SQLite is exploited to persist the information on the local storage of the device. Java Native Interface (JNI) is used to establish the communication between the vision module and the application. Data-interchange is achieved through JSON messages which contain data about artwork detected and recognized, persons in the field of view of the camera, presence of speech as well as user movements found through visual information.

Experimental Results

Dataset. To train and evaluate our object detector described in Sect. 1.4, and later evaluate the full recognition system, we collected a dataset of artworks situated in the Bargello Museum in Florence. Artworks data have been collected in form of $1,237$ images taken from live footage of the museum under diverse illumination and viewpoint conditions. Changes in these conditions produce a significant alteration of artwork appearance mainly due to the sensitivity of camera sensor and the diverse behavior of light reflection of different surfaces.

To comply with object detection task, we annotated a subset of images with information about the pictured artworks and split them into training and testing sets. To balance the artwork and person classes the latter has been augmented with images from PASCAL VOC2007, resulting in ~ 300 person and ~ 300 artwork images. Data augmentation techniques have also been applied, such as image flipping.

We evaluate our recognition system with an additional set of images annotated with the correct artwork id. To produce this set a tool has been developed that uses our pipeline by employing both the artwork detector to generate bounding boxes and a tracker, to aggregate boxes belonging to the same sequence. Such sequences have than been manually annotated, assigning each one to a new or existing artwork id. As already discussed in Sect. 1.4, thanks to the non-parametric nature of the recognition system, this process can be repeated multiple times to further enrich the dataset. Special attention has been put to include examples with multiple artworks.

The resulting dataset consists of sequences accounting for $8,820$ frames, where each visible artwork is annotated with its corresponding label and bounding box coordinates. All the sequences account for a total of ~ 250 seconds of video with $7,956$ detections.

Artwork detection. We evaluate the performance of the artwork detection system as first experiment. After fine-tuning the network on our dataset we measure the detector average precision on the test set. As described in Sec. 1.4, since we are interested in detecting the artwork in front of the user, we limit the results to those detections for which the detected area, normalized to the dimension of the frame, is at least T, and report the average precision when varying its value. The experiment shows that increasing the minimum area of a box yields and increase of the average precision, that reach its peak of 0.9 at 40% of the area. This means that the classifier is more effective at recognizing artworks close to the user. Increasing the minimum area is also not a guarantee of better results, in fact while far detections are prone to errors due to small scale, errors can also arise at close distance due to blur effect of the camera.

A good value for T must therefore be a trade-off between good precision and how close an artwork must be to the user. We choose a value of $T = 0.1$, that provides a significant improvement in precision with respect to the bare detector at a distance of ~ 5 meters. Fig. 1.8 reports the precision-recall curve relative to the chosen T, showing very good precision at high recall rates and exhibiting only a small decrease

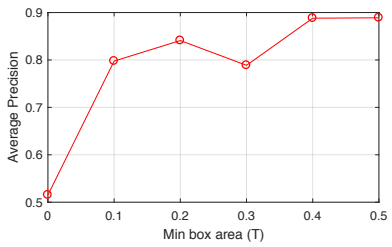

Figure 1.7: Average precision of artwork detection varying the minimum box area T.

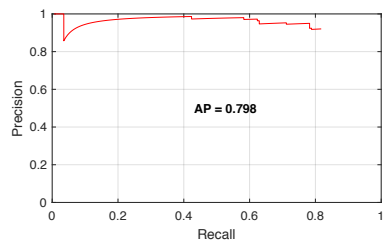

Figure 1.8: Precision-recall curve for artwork detection using a threshold $T = 0.1$.

in precision until ~ 0.8 recall. Note that, according to the results reported in Fig. 1.7 for the selected threshold, a higher recall cannot be obtained and therefore the curve in Fig. 1.8 is truncated at that point.

Artwork recognition: nearest neighbour evaluation. With the following experiment we aim at first evaluating the descriptor fusion approach described in Sect. 1.4 to find the best pair of layers, and then evaluate how varying the number of nearest neighbours affects in terms of precision the artwork recognition system when using the best configuration. Fig. 1.9 reports the recognition accuracy of the combination of layer 3 and 4 with all the other convolutional layers, showing that fusing the 3^{rd} and 4^{th} layers yields the best results. Using this configuration we then vary the number of neighbours and report in Fig. 1.10 the variation in accuracy, observing that using just 1 nearest neighbor provides the best performance in recognizing an artwork. Using more neighbours to vote the correct artwork id does not improve the accuracy. The reason for this behaviour is that for each artwork we acquired multiple poses in different lighting conditions, so for each query there will be only a small number of samples in a similar pose with a similar lighting condition. Increasing the amount of neighbours simply adds noisy data to the results.

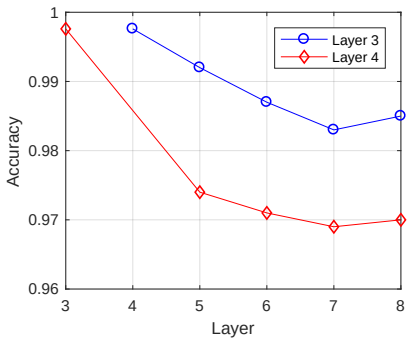

Figure 1.9: Recognition accuracy of combinations of layer 3 and 4 with layers $[3, \ldots, 8]$.

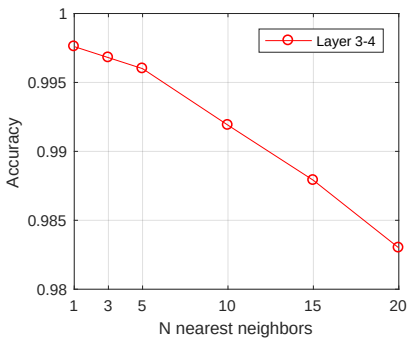

Figure 1.10: Recognition accuracy of the best layer combination 3+4, varying number of NN.

Test	Strategy			Correct	Incorrect	Skipped
	C	D	P			
T1	✗	✗	✗	5,598 (~70%)	2,358 (~30%)	0 (0%)
T2	✗	✓	✗	5,334 (~67%)	1,267 (~16%)	1,355 (~17%)
T3	✓	✗	✗	4,475 (~56%)	36 (~0%)	3,445 (~43%)
T4	✓	✓	✗	4,363 (~55%)	11 (~0%)	3,582 (~45%)
T5	✓	✗	✓	5,141 (~65%)	61 (~1%)	2,754 (~35%)
T6	✓	✓	✓	4,966 (~62%)	22 (~0%)	2,968 (~37%)

Table 1.1: **Performance by applying the three strategies for temporal smoothing: C**onsistency, **D**istance and **P**ersistence. We report the number of detections where, respectively, the artwork was: correctly recognized, misclassified and generically labeled as "artwork".

Temporal Processing Evaluation. Here we test the effectiveness of the three temporal processing strategies, i.e. *Consistency*, *Distance* and *Persistence*, described in Sec. 1.4, by experimenting with several of their combinations. Using a simulation of the system where we set $T = 0.1$, $M = 15$ and with a persistence of artwork detection of $P = 20$ recognitions, we evaluate the annotated video sequences tracking every combination of output bounding box and label, comparing the result to the ground truth. The number of correctly, incorrectly and generically labeled as "artwork" detections is used as performance measure and reported in Table 1.1.

We perform 6 tests labeled from T1 to T6. In T1, where we do not apply any strategy to provide a baseline for the other tests, we can observe that only 70% of the labels are correctly classified, leaving 30% of incorrect labels. By applying the *Distance* criterion in T2 we halve the number of incorrect labels but at the expense of assigning a generic label to the the other half, confirming that a large number of errors is made on distant, and therefore difficult to recognize, artworks. In T3 we evaluate the *Consistency* strategy which shifts almost all incorrect labels and around 15% of the correct ones to the generic label. This is due to the video system not providing a stable output label among consecutive frames.

In T4 we combine *Distance* and *Consistency* criteria, noting that, while *Consistency* is able to almost nullify the incorrect recognitions, it is not robust to sparse errors, causing the system to often bounce from the correct recognition to the generic label. This issue is resolved in T5 where *Consistency* is combined with *Persistence*, leading to a reduction in misclassified labels, an increase in the number of correct labels and a decrease of generic ones. Finally in T6 we combine all three criteria. Here we can see how using all the strategies together produces only 22 incorrect labels, which approximately corresponds to less than one cumulative second every $\sim$ 5 minutes of video, at the expense of a reasonable number of generically classified artworks. This also confirms our intuition about the efficacy of the three strategies.

Voice Detection Evaluation

Here we evaluate two strategies for voice activity detection: *i)* Sample, we evaluate one audio sample per second; and *ii)* Mean, we evaluate the mean sample computed

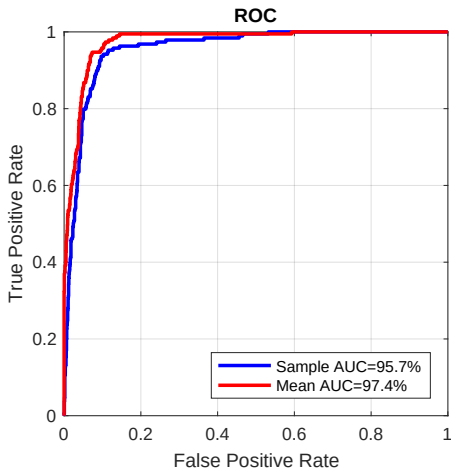

Figure 1.11: Receiver operating characteristic curve of the tested voice activity classifiers.

over a second of audio. To minimize the number of false positives we measure the classifier performance varying the false positive threshold.

In Fig. 1.11 we report the Receiver Operating Characteristic (ROC) curve. Although both strategies have a high Area Under the Curve (AUC) and correctly classify voice most of the time, the Mean strategy performs always better then Sample.

1.5 THE OUTDOOR SYSTEM

The system, for which the architectural diagram is shown in Fig. 1.13, is composed of three interacting modules: *i)* a *Location Module* to provide current location and nearby interest points; *ii)* a *Content Provider* responsible to fetch textual information; and *iii)* a *Vision Module* constantly comparing the user point of view to a set of expected interest point appearance images.

The system has been developed as an Android Application and makes use of the following sensors: a *GPS receiver* to perceive the user position, and a *camera* to validate the artwork he faces. The Application is responsible of coordinating the three main modules and to present the final information to the user (see Fig. 1.12).

Context Awareness

For a fully autonomous audio guide understanding the wearer context must be considered of paramount importance if meaningful information is to be provided at the right moment. The simple fact that the user is facing an interest point does not always mean that he is interested in it because he could also be enjoying other activities. For this reason we also use the device sensors, as explained in Sec. 1.5, to estimate the

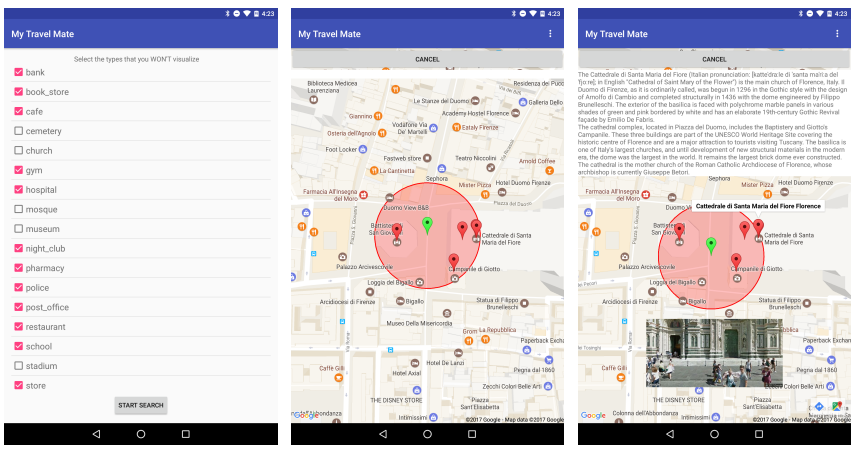

Figure 1.12: *(left)* the application is asking the user the interest point types he does not want to be included in the search, *(center)* the application is showing the interest point around the user, *(right)* the application shows the textual description for the selected point

amount of interest of the user, differentiating simple wandering from observing.

Application Modules

Location Module This module is responsible for providing interest points by exploiting the current user position and querying the Google Place Search API[3] for a given radius. The response is a list of 60 interest points each one annotated with one or more `type`[4], such as `art_gallery` or `bank`. The user can personalize the application by specifying which type he is interested into (Fig. 1.12, left). The interest points are then passed to the *Content Provider* to retrieve textual descriptions, together with the name of the city the user is visiting which is obtained by performing a Geocoding[5] query. This is necessary to discriminate interest points with the same name but located elsewhere.

Content Provider This module fetches textual information for interest points found by the *Location Module* by taking the first result of a Wikipedia[6] search in the local language for articles containing both the city name and the interest point name. The description is then obtained with another Wikipedia[7] search, limiting the description text to the page summary. To provide translations into other languages the module performs an additional query to Wikipedia requesting the Interlanguage

[3] https://developers.google.com/places/web-service/search
[4] https://developers.google.com/places/supported_types
[5] https://developers.google.com/maps/documentation/geocoding
[6] https://www.mediawiki.org/wiki/API:Search
[7] https://www.mediawiki.org/wiki/API:Query

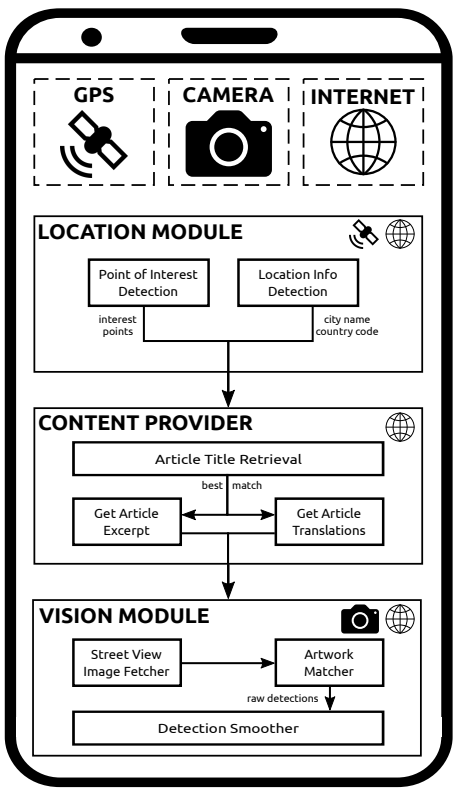

Figure 1.13: System Architecture

Links[8] for the retrieved page and downloading the extracts for the desired languages. If an interest point name does not match any Wikipedia article or the content is not available for the desired language, then the point is discarded.

Vision Module Understanding when the user is correctly facing a landmark is not trivial. As shown in Fig. 1.14, GPS position and device orientation are not reliable information and cannot be used directly. We can observe that GPS coordinates do not accurately follow the path of the visitor, possibly because GPS satellites are often obscured by tall buildings. Moreover, the device orientation, as reported by sensors, is also inaccurate. The *Vision Module* addresses these problems by using a computer vision algorithm that constantly observes the user perspective and matches it to the surrounding artworks provided by the *Location Module*. Matching is done against Google Street Map images picturing the interest points from the user position. For each landmark we retrieve 4 views by varying the image angle of ±10

[8] https://en.wikipedia.org/wiki/Help:Interlanguage_links

degrees and the pitch of ±5 degrees thus building a small localized dataset.

To keep the vision pipeline efficient, we extract SIFT features from the small dataset and the frame coming from the camera and match them as described in [29]. If enough matches are found against one of the dataset images, the interest point is recognized as the one in the image. To reduce the computational cost the *Vision Module* analyzes only a frame every 0.5s (i.e. 2 detections/second).

Temporal Smoothing

The *Vision Module* does not provide instantaneous results for each input frame, instead it needs to process a sequence of frames before outputting a decision. This allows us to perform a post-processing over single outputs and present a smoothed output to the user. The module evaluates each frame and considers the output *valid* if it persists for at least 5 frames; moreover, a Time To Live (TTL) of 5 frames is also applied to tolerate sporadic incorrect results. The TTL is decremented whenever a decision differs from the valid one and is reset in case of matching decision. If TTL reaches zero all invalid frames are considered incorrect, otherwise they are all assigned to the artwork of the valid frames. This behavior allows the module to tolerate small classification noise. In addition, to enforce the production of long lasting sequences, the modules reports the valid output only if it lasts at least for 10 frames.

Exploiting Sensors for Modeling Behavior

In an outdoor scenario where artworks are often whole buildings, moving does not necessarily implies disinterest. We monitor GPS data and compute user distance from the artwork to understand if he moves but stays close to the artwork, thus continuing to be interested in it, or if he moves away.

This spacial information is used to make the audio persist when the *Vision Module* stops detecting an artwork. This is crucial in outdoor scenarios where occlusion is a very frequent event, so once the artwork is recognized the description is provided to the user as long as he stays close to it or a different artwork is recognized by the *Vision Module*. In case he walks away and no other artwork is recognized then audio is stopped.

System Implementation

As for the indoor system, the vision system introduce in Sec. 1.5 has been developed and tested using an NVIDIA Jetson TX1 board. The audio-guide application has been deployed on an NVIDIA Shield Tablet K1 which has the same hardware of the TX1 board but runs Android 6.0 instead of Linux. The mobile application has been developed using the Android SDK and makes use of Java Native Interface (JNI) to communicate with the *Vision Module*. Data between Java and Native Code is exchanged using JSON messages providing information on the detected artworks.

Application Use Cases

The application has been developed with the intention of letting the user enjoy his visit without needing to interact with the guide. Although this is the primary behavior, the application can also work in a non automated way, giving the user the ability to virtually explore his surrounding. These modalities defines two different use cases: *i*) a *Fully Automated* mode and *ii*) an *Interactive* mode.

Fully Automated In this scenario the user walks freely with the device positioned in a front pocket to let the camera look at what is in front of him. Meanwhile, the application monitors the user GPS position and looks for interest points through the *Location Module*. When the user walks in a zone where interest points are available, the *Content Provider* fetches information about them and the *Vision Module* analyzes the camera stream to understand if the visitor is facing one of them. In case of positive match, the application delivers the downloaded information to the user as audio description.

Interactive This modality is available to the user to let him virtually explore his surroundings. The user can move to any position on the map, as shown in Fig. 1.12, to discover interest points. By touching one of them the application will show the user the relative textual description, which can be then transposed to audio if desired. In this modality the *Vision Module* is paused.

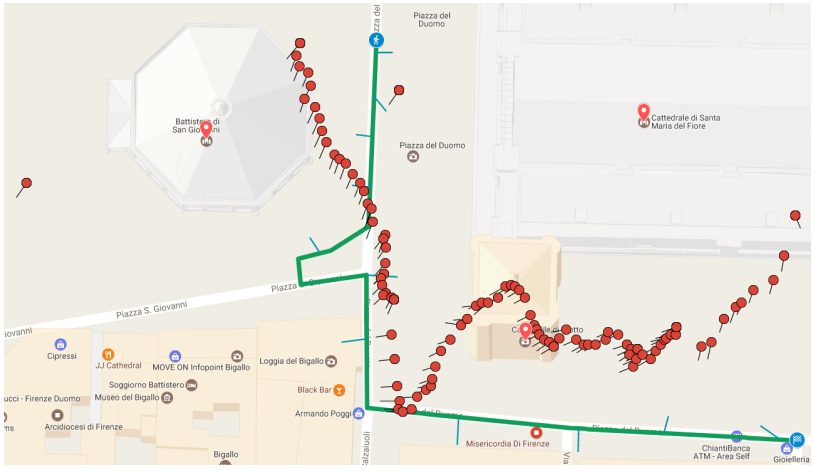

Figure 1.14: Map of the Duomo Square of Florence. The three points of interest (red markers) are highlighted together with the user location points obtained from GPS (red circles) and the real tourist itinerary (green line). The real tourist orientation and the one obtained from GPS are marked, respectively, with green lines along the itinerary and black lines on the red circles.

Experimental Results

Dataset. To evaluate the application vision system we collected a small dataset of video footage of city center of Florence. The city of Florence has a rich architectural and artistic history and its center hosts a great variety of artworks. The footage records the visit of 3 buildings, filmed from different points of view and different lighting conditions, for a total of 5 minute of video. Every 0.5 seconds we annotated the current frame with time of the visit and the GPS position, resulting in 522 annotated frames.

The evaluation is carried by selecting 4 points during the visit and for each of them we queried the Google Street View API to download images of the 3 interest points and obtain their appearances from those positions. To account for sensor inaccuracy we slightly varied the *heading* and the *pitch* of $\pm 10°$ and $\pm 5°$ respectively, obtaining 4 different images of the same artwork.

Method Evaluation. We evaluate the system as follows: for each of the 522 GPS landmarks we extract SIFT descriptors from the relative frame and match them to the ones extracted from the reference images in the dataset. If enough matches are found between two images then we label the current frame as containing the artwork pictured in the matching reference image. After this process we obtain a sequence of 522 descriptors labeled either with one of the artworks or as background.

Performance is evaluated in terms of accuracy, comparing the predicted label series to the ground-truth. In Fig. 1.15 we report the ground-truth (left) and the results, showing both the raw output (center) of the module and the smoothed one (right) as explained in Sec. 1.5. The center image confirms the erratic behavior of the results when no filtering is applied, producing a quite low accuracy of 70%. Applying filtering, accuracy increases considerably by 20% proving that the smoothing is necessary. It can be noted that filtered results and ground-truth differ slightly only on the starting and ending points of the detections. This means that we always give the correct output label but we make a small error in detecting the exact starting and ending landmarks. It is worth noting that perfect timing is not essential for an audio guide application, in fact it is preferable to give the correct audio description with a short delay than giving a wrong description at the correct time.

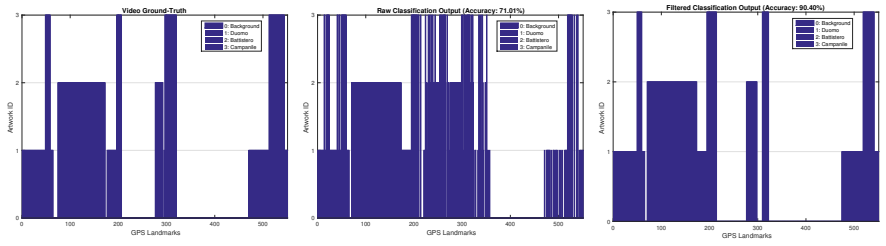

Figure 1.15: *(left)* ground truth ids for video GPS landmarks, *(center)* raw predictions for each video GPS landmark, *(right)* final prediction output of the method for each video GPS landmark

Type	Instrumental	Social	Interactional	Communicative
Human Guide	* * *	* * *	* * *	* * *
Audio Guide	*	–	–	* *
Our System	* *	* *	* *	* *

Table 1.2: Functions comparison of our guide with respect to human and traditional audio guides

User experience evaluation

Human guide roles have been connoted in [9] through a distinction between their instrumental (guide), social (animator), interactional (leader) and communicative (intermediator) functions. Instrumental functions convey information such as artwork localization and user routing. Interactional aspects provide users with means of getting information. Sociality involves user engagement, also supplying collaborative experiences. Communicative functions facilitate personalisation and ease access to targeted content. Although humans are the best in carrying out these functions, recent progresses in technology have given the opportunity to achieve close results. In Table 1.2 a qualitative comparison is shown between functions provided by human and traditional audio-guides with those achievable by our system.

Our audioguide presents appreciable improvements especially as regards to instrumental, interactional and socialization aspects as: 1) it allows the least cognitive effort to obtain information, which is provided automatically allowing at the same time playback control; 2) it senses the context and understand user behaviour adapting accordingly (e.g. stopping when the user loses attention), avoiding the 'isolated experiences' provided by traditional audio-guides. The overall experience achieved by the system in a real environment was measured through an evaluation of its usability both for the outdoor and the indoor scenario. The popular Standard Usability Scale (SUS) [7] was used, following a user-centred approach. Different groups of users were asked to fulfil some tasks in both the scenarios and then to respond to a 10 points questionnaire. Answers to questions are based on a Likert scale [41] from 1 to 5, expressing a range between "Strongly Disagree" and "Strongly Agree" with opposite meaning, a strategy proved effective in minimizing acquiescence bias, and alternating positive with negative sentences so that users have to read carefully. As stated in [33], it is sufficient to collect 5 polls to detect the 85% of design errors of a user interface. For both the indoor and the outdoor system a supervised and an unsupervised scenario (i.e. receiving or not explanations on the app functionalities and on the recognition system) were evaluated. Users had to get information on one or more artworks/landmarks of interest. The indoor system was tested by 12 users and obtained an average SUS of 74.0 for the unsupervised scenario and 79.5 for the supervised scenario; the outdoor system was tested by 10 users with an average SUS of 69.5 for the unsupervised scenario and 71.8 for the supervised scenario. Results, quite close to 80 for the first scenario, denote an interface providing a very good ex-

perience; while for the second, which reached a score between the reference values of 60 and 80, the user experience was above average [37]. The small difference in supervised and unsupervised scores suggests that the system is easy to use also without previous training in both scenarios. Feedback, gained through open-ended questions on system usability, which, as stated by Nielsen, allow 'to find out more than you anticipate' are summarised as follows: 11 users indicated the automatic recognition of artworks (i.e. 6 users) and the consequent start/stop of the audio reproduction (i.e. 5 users) as the best feature. Issues were highlighted by 2 users: they are relative to the use of a system as a mobile app in assisted mode and regard the fatigue caused by having to constantly hold the device in front of the artworks with the hands.

1.6 CONCLUSIONS

New devices featuring powerful GPUs for building and deploying embedded systems applications, like the NVIDIA Jetson TX1 and on NVIDIA Shield Tablet K1, give the possibility to design and implement new tools, such as the smart audio-guide presented in this Chapter, that can considerably improve and change consolidated paradigms for the fruition of multimedia insights in the context of museum and cultural experiences.

The presented audio-guide system allows to profile the visitor in real-time, understanding his interests and the context of the visit, and to provide natural mechanism of interaction and feedback for the attainment and the semi-automatic control of multimedia insights. Audio and sensor data are exploited to improve the user experience reducing the difficulty of use and the cumbersome approach of a traditional audio-guide. In proposing our method we discuss the opportunity to adopt different techniques for performing artwork recognition in the two main scenarios of an outdoor and indoor visit (i.e. CNN and SIFT). Furthermore, the proposed solution show some good practices for the fusion of data (visual, acoustic, motion and localization) coming from different types of sensors on wearable mobile devices and suggests simple but effective temporal smoothing and consistency strategies. These strategies allow to obtain a lower error rate in automatic artwork recognition and an overall improved human computer interaction.

The smart audio-guides have been implemented as Android apps providing the users with the possibility to switch between a fully automated experience to a semi-automatic, more interactive mode where the system is not worn but used as a traditional application on a mobile device through an *ad hoc* configurable interface.

Usability testing, performed with the commonly used SUS (System Usability Scale), has given good results for both scenarios, proving that understanding the context can help to provide an improved experience to the user.

REFERENCES

[1] Gregory D Abowd, Christopher G Atkeson, Jason Hong, Sue Long, Rob Kooper, and Mike Pinkerton. Cyberguide: A mobile context-aware tour guide. *Wireless networks*, 3(5):421–433, 1997.

[2] Rao Muhammad Anwer, Fahad Shahbaz Khan, Joost van de Weijer, and Jorma Laaksonen. Combining holistic and part-based deep representations for computational painting categorization. In *Proc. of ICMR*, 2016.

[3] Liliana Ardissono, Tsvi Kuflik, and Daniela Petrelli. Personalization in cultural heritage: the road travelled and the one ahead. *User Modeling and User-Adapted Interaction*, 22(1):73–99, 2012.

[4] Lorenzo Baraldi, Francesco Paci, Giuseppe Serra, Luca Benini, and Rita Cucchiara. Gesture recognition using wearable vision sensors to enhance visitors' museum experiences. *IEEE Sensors Journal*, 15(5):2705–2714, 2015.

[5] Jonathan Bowen, James Bradburne, Alexandra Burch, Lynn Dierking, John Falk, Silvia Filippini Fantoni, Ben Gammon, Ellen Giusti, Halina Gottlieb, Sherry Hsi, et al. *Digital technologies and the museum experience: Handheld guides and other media*. Rowman Altamira, 2008.

[6] Jonathan P Bowen and Silvia Filippini-Fantoni. Personalization and the web from a museum perspective. In *Proc. of Museums and the Web (MW)*, 2004.

[7] John Brooke. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, pages 189–194, 1996.

[8] Luca Caviglione, Mauro Coccoli, and Alberto Grosso. A framework for the delivery of contents in rfid-driven smart environments. In *Proc. of RFID-TA*, 2011.

[9] Erik Cohen. The tourist guide: The origins, structure and dynamics of a role. *Annals of Tourism Research*, 12(1):5–29, 1985.

[10] Alberto Del Bimbo, Andrea Ferracani, Daniele Pezzatini, Federico D'Amato, and Martina Sereni. Livecities: revealing the pulse of cities by location-based social networks venues and users analysis. In *Proc. of WWW*, 2014.

[11] Alberto Del Bimbo, Walter Nunziati, and Pietro Pala. David: Discriminant analysis for verification of monuments in image data. In *Proc. of ICME*, 2009.

[12] Thomas Drugman, Yannis Stylianou, Yusuke Kida, and Masami Akamine. Voice activity detection: Merging source and filter-based information. *IEEE Signal Processing Letters*, 23(2):252–256, 2016.

[13] Veron Eliseo and Levasseur Martine. Ethnographie de l'exposition. *Études et recherche, Centre Georges Pompidou, Bibliothèque publique d'information*, 1991.

[14] Benjamin Elizalde and Gerald Friedland. Lost in segmentation: Three approaches for speech/non-speech detection in consumer-produced videos. In *Proc. of ICME*, 2013.

[15] Christos Emmanouilidis, Remous-Aris Koutsiamanis, and Aimilia Tasidou. Mobile guides: Taxonomy of architectures, context awareness, technologies and applications. *Journal of Network and Computer Applications*, 36(1):103–125, 2013.

[16] Florian Eyben, Felix Weninger, Stefano Squartini, and Björn Schuller. Real-life voice activity detection with LSTM recurrent neural networks and an application to hollywood movies. In *Proc. of ICASSP*, 2013.

[17] Ross Girshick. Fast R-CNN. In *Proc. of ICCV*, 2015.

[18] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jagannath Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proc. of CVPR*, 2014.

[19] Luca Greci. An augmented reality guide for religious museum. In *Proc. of AVR*, 2016.

[20] Loc Nguyen Huynh, Rajesh Krishna Balan, and Youngki Lee. Deepsense: A GPU-based deep convolutional neural network framework on commodity mobile devices. In *Proc. of WearSys*, 2016.

[21] Svebor Karaman, Andrew D. Bagdanov, Lea Landucci, Gianpaolo D'Amico, Andrea Ferracani, Daniele Pezzatini, and Alberto Del Bimbo. Personalized multimedia content delivery on an interactive table by passive observation of museum visitors. *Multimedia Tools and Applications*, 75(7):3787–3811, 2016.

[22] Michael Kenteris, Damianos Gavalas, and Daphne Economou. Electronic mobile guides: a survey. *Personal and ubiquitous computing*, 15(1):97–111, 2011.

[23] Doyeon Kim, Daeil Seo, Byounghyun Yoo, and Heedong Ko. Development and evaluation of mobile tour guide using wearable and hand-held devices. In *Proc. of HCI*, 2016.

[24] Ronny Kramer, Marko Modsching, and Klaus Ten Hagen. A city guide agent creating and adapting individual sightseeing tours based on field trial results. *International Journal of Computational Intelligence Research*, 2(2):191–206, 2006.

[25] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. of NIPS*, 2012.

[26] Tsvi Kuflik, Zvi Boger, and Massimo Zancanaro. *Analysis and Prediction of Museum Visitors' Behavioral Pattern Types*, pages 161–176. Springer Berlin Heidelberg, 2012.

[27] Seyyed Salar Latifi Oskouei, Hossein Golestani, Matin Hashemi, and Soheil Ghiasi. CNNdroid: GPU-accelerated execution of trained deep convolutional neural networks on android. In *Proc. of ACM MM*, 2016.

[28] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: Single shot multibox detector. In *Proc. of ECCV*, 2016.

[29] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[30] Austin Meyers, Nick Johnston, Vivek Rathod, Anoop Korattikara, Alex Gorban, Nathan Silberman, Sergio Guadarrama, George Papandreou, Jonathan Huang, and Kevin P. Murphy. Im2calories: Towards an automated mobile vision food diary. In *Proc. of ICCV*, 2015.

[31] Ananya Misra. Speech/nonspeech segmentation in web videos. In *Proc. of Interspeech*, 2012.

[32] Saman Mousazadeh and Israel Cohen. Ar-garch in presence of noise: parameter estimation and its application to voice activity detection. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(4):916–926, 2011.

[33] Jakob Nielsen and Rolf Molich. Heuristic evaluation of user interfaces. In *Proc. of CHI*, 1990.

[34] David Picard, Philippe-Henri Gosselin, and Marie-Claude Gaspard. Challenges in content-based image indexing of cultural heritage collections. *IEEE Signal Processing Magazine*, 32(4):95–102, 2015.

[35] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proc. of CVPR*, 2016.

[36] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Proc. of NIPS*, 2015.

[37] Jeff Sauro and James R. Lewis. *Quantifying the user experience: Practical statistics for user research*. Morgan Kaufmann, 2012.

[38] Todd Simcock, Stephen Peter Hillenbrand, and Bruce H Thomas. Developing a location based tourist guide application. In *Proc. of ACSW Frontiers*, 2003.

[39] Yuichiro Takeuchi and Masanori Sugimoto. A user-adaptive city guide system with an unobtrusive navigation interface. *Personal and Ubiquitous Computing*, 13(2):119–132, 2009.

[40] Ryosuke Tanno, Koichi Okamoto, and Keiji Yanai. Deepfoodcam: A DCNN-based real-time mobile food recognition system. In *Proc. of MADiMa*, 2016.

[41] William M. Trochim et al. Likert scaling. *Research methods knowledge base, 2nd edition*, 2006.

[42] Fabio Vesperini, Paolo Vecchiotti, Emanuele Principi, Stefano Squartini, and Francesco Piazza. Deep neural networks for multi-room voice activity detection: Advancements and comparative evaluation. In *Proc. of IJCNN*, 2016.

[43] Yafang Wang, Chenglei Yang, Shijun Liu, Rui Wang, and Xiangxu Meng. A rfid & handheld device-based museum guide system. In *Proc. of ICPCA*, 2007.

[44] Yiwen Wang, Natalia Stash, Rody Sambeek, Yuri Schuurmans, Lora Aroyo, Guus Schreiber, and Peter Gorgels. Cultivating personalized museum tours online and on-site. *Interdisciplinary Science Reviews*, 34(2-3):139–153, 2009.

[45] Kyoung-Ho Woo, Tae-Young Yang, Kun-Jung Park, and Chungyong Lee. Robust voice activity detection algorithm for estimating noise spectrum. *Electronics Letters*, 36(2):180–181, 2000.

[46] Jiaxiang Wu, Cong Leng, Yuhang Wang, Qinghao Hu, and Jian Cheng. Quantized convolutional neural networks for mobile devices. In *Proc. of CVPR*, 2016.

[47] Keiji Yanai, Ryosuke Tanno, and Koichi Okamoto. Efficient mobile implementation of a CNN-based object recognition system. In *Proc. of ACM MM*, 2016.

[48] Massimo Zancanaro, Tsvi Kuflik, Zvi Boger, Dina Goren-Bar, and Dan Goldwasser. Analyzing museum visitors' behavior patterns. In *Proc. of UM*, 2007.