UNIVERSITÀ
DEGLI STUDI
FIRENZE

Università degli Studi di Firenze
Dipartimento di Ingegneria dell'Informazione (DINFO)
Corso di Dottorato in Ingegneria dell'Informazione

Curriculum: Informatica

Exploiting Digital Traces and Human Factors to Design and Improve HCI Systems for Online, Outdoor and Indoor Environments.

Candidate
Andrea Ferracani

Supervisor
Prof. Alberto Del Bimbo

PhD Coordinator
Prof. Enrico Vicario

ciclo XXX, 2014-2017

Università degli Studi di Firenze, Dipartimento di Ingegneria
dell'Informazione (DINFO).

Thesis submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Information Engineering.

A mio padre

Acknowledgments

I would like to acknowledge the efforts and input of my supervisor, Prof. Alberto Del Bimbo and all my colleagues of the MICC (Media Integration and Communication Center) who were of great help during my research.

Contents

Chapter 1

Introduction

1.1 The objective

This PhD thesis deals with the study of new paradigms for exploiting data obtained from virtual and real digital traces (i.e. from Social Networks, Location-Based Social Networks, remote sensing and crowd-sourcing, physic sensors) and human factors (i.e. profiling, behavioral aspects such as movements, voice and contextual information) as a basis for the design and implementation of innovative infovis technologies, multimedia recommendation and browsing systems, human computer interaction paradigms in real and virtual spaces, i.e. in online, outdoor and indoor environments. Primary objectives have been the automatic identification of functional areas within cities through the analysis of geolocated social media information and user profiling, and the detection of macro and micro events happening in urban areas. The analysis on user generated data and the profiling of user behaviours, actual such as movements and voice interaction and implicit or psychological such as self-esteem and saliency perception have also led to research on how to exploit user profiling on online platforms and human factors to improve recommendations systems on general purpose and contextual social networks. The same approach has also been used as regard to real locations in designing smart and personalized systems to be used in indoor and virtual environments.

1.2 Contributions

New methods for identification of functional areas in cities and for the detection of geo-localized micro and macro-events are described respectively in Chapter 2 and Chapter 3. Using these informations as a basis, personalized recommendation and routing systems as well as immersive browsing interfaces has been created that could present innovative paradigms of human computer interaction through original solutions for information visualizations. In particular it has been developed 1) a framework which exploits user profiling on social networks and data from mobile device sensors that allows the semi-automatic recommendation and assisted definition of personalized itineraries, described in Chapter 4a nd 2) a framework for navigating, on the web, immersive urban routes through continuous spherical images transition, which allows the fruition of multimedia materials associated to Point-Of-Interests, described in Chapter 5. These systems exploit in part the techniques of user profiling presented in the first Chapter and propose and provide original solutions for information visualization. Proposed strategies of recommendation of multimedia material combined with the previously cited user profiling techniques have then been used for also proposing 1) a recommender which makes an improvement to standard user and item-based recommendation algorithms, described in 6. The recommender has been implemented in a Social Network for video sharing and it is based on content-based techniques for the analysis of video annotations (the improvement has been achieved through an original exploitation of human factors, i.e. users' self-expression in user profiles and perception of visual saliency in video frames); 2) effective methods of recommendation and profiling designed for contextual SNs which exploits online profiling, collaborative filtering and information derived from the context, cf. Chapter 7. New ways of recommending, browsing and navigating multimedia content in indoor real and virtual environments relying on context understanding, behaviour and interests profiling, user localisation through portable devices, ibeacon sensors and computer vision, and natural interaction have been designed and implemented in real applications in original ways. Insights from these experiences and solutions are described in Chapter 8, 9 10, 11 and 12. Main objectives of these systems were the personalization of content and the understanding of the user context, preferences and behaviors in order to provide a more natural and targeted user experience.

1.3 Digital Traces

Physical traces that humans leave on things in the world are everywhere. They can be visible or imperceptible, left intentionally or unintentionally. Just think of footprints on a floor, a lost hair in the bed or between the pillows of the couch, or a note that we have written and abandoned somewhere to remind us of something. As regard to invisible traces, we can consider the fingerprints that we leave on anything we touch with bare hands. All these traces are essential information that reveal where we have been, how we got there, objects we have handled, with whom we have been, and they can say a lot even with respect to our interests and preferences. This is no different in the digital world, where it happens on an even larger scale. We all, aware or unaware, leave hundreds and hundreds of digital traces every day working at our computers (sending emails, writing blog posts, interacting on social networks through Twitter messages, Facebook status updates, Youtube videos views and uploads, etc.) or simply carrying around our mobile phones with us and using applications (through logs of device sensor data such as GPS, accelerometers, camera, microphone etc.). Through these logs it is possible to extract information that we would assume to be private and that can instead give detailed insights of our lives that are revealed unintentionally (as it happens for example through the records of website visits and queries on search engines). Digital traces can be defined as metadata, i.e. data that describe the content itself so that its semantic value can be contextualized. Whether we consider the metadata produced using our personal devices, for example by writing a message on a social network that can be enriched through geolocation, timestamp, message content analysis; or those that identify our actions captured by external devices, such as a video-surveillance camera, or worn devices in a precise environment (e.g. exploiting localization, user attention and preference estimation through computer vision, voice recognition, activity detection), all these are information that have to do a lot with who we are, how we represent ourselves in online or virtual worlds or that just reveal what we are doing and how we act in public.

Today much of our activities, professional and private, are mediated by information systems. This means that the massive amount of digital footprints that are produced can be used to gain insights on people behaviors and to adapt technologies to personal preferences and needs. This can be done both at the macroscopic and microscopic levels and using the different paradigms in which digital technologies can be exploited, i.e. in online, outdoor, indoor or virtual environments. With macroscopic level we mean, for example, to process user-generated information in order to understand social and urban dynamics, or to detect and to predict events that could be of interest for the community itself or, for example, for city planners. With microscopic level instead we reference all those systems for the fruition and recommendation of content that take into account personal traces and are directed specifically to the use of individuals in ‘spaces’. In this context a new breed of interactive systems can be designed that can help to bridge the gap between users of applications, the multimedia content and the world itself where users move in. These systems can be used through our personal devices in online environments, where our digital representations are nowadays the mirror of the real ones (think about our social network profiles), but also in outdoor and indoor environments, or exploiting new technologies such as virtual reality systems and natural interaction.

An important aspect discussed in this thesis is the investigation of how digital traces can be exploited at their best considering at the same time user preferences and behaviors, the context of use of technologies and their objective, that can be of the most varied. Human factors are in fact related to the context of use and come into play on a personal and social level when people use digital systems. Through several examples it is demonstrated how user interests, perception, self-esteem, user status and conditions, contextual needs, in particular and located situations, can be a means to better define digital traces and to improve digital systems in terms of targeted services, usability of applications and users expectations.

The thesis has 12 chapters, which are briefly summarized. Chapter 2 and 3 are dedicated to the exploitation of large amount of personal data from social networks to extract knowledge that can be valuable, at the macroscopic level, to have insights on urban dynamics. **Chapter 2** presents a system for the automatic detection of functional areas in cities using geolocated data from Facebook. The main idea is to characterize points of interests in cities through the analysis of profiles of people who have been there. It is demonstrated how this technique in feature selection improves the detection compared to standard approaches.

Chapter 2

The role of user profiling in detecting functional areas in cities

It would be very difficult even for a resident to characterise the social dynamics of a city and to reveal to foreigners the evolving activity patterns which occur in its various areas. To address this problem, however, large amount of data produced by location-based social networks (LBSNs) can be exploited and combined effectively with techniques of user profiling. The key idea introduced in this Chapter is to improve city areas and venues classification using semantics extracted both from places and from the online profiles of people who frequent those places. Results of this methodology are presented in LiveCities, a web application which shows the hidden character of several italian cities through clustering and information visualisations paradigms. In particular in-depth insights are given for the city of Florence, IT, for which the majority of the data in the dataset have been collected. The system provides personal recommendation of areas and venues matching user interests and allows the free exploration of urban social dynamics in terms of people lifestyle, business, demographics, transport etc. with the objective to uncover the real 'pulse' of the city. We conducted a qualitative validation through an online questionnaire with 28 residents of Florence to understand *the shared perception of city areas by its inhabitants and to check if their mental maps align to our results. Our evaluation shows how considering also contextual semantics like people profiles of interests in venues categorisation can improve clustering algorithms and give good insights of the endemic characteristics and behaviours of the detected areas.* [1]

2.1 Introduction

An analysis capable to convey to a realistic and truthful representation of a city and of the activities that take place in its areas must necessarily take into account not only human mobility but also users' socio-economic characteristics and interests distribution. Emerging social realtime systems offer an opportunity for the computation in the field of spatial data mining due to the huge amount of geo-localised data they continuously produce and that can be considered real human sensor data.

There exist a considerable number of works addressing geographical modelling of information derived from widespread LBSNs like Twitter and Foursquare. Some recent studies analyse social media streams to obtain contextual semantics for city zones and venues whilst others focus more on human mobility. In [65] user's positions are observed predicting the locations of new tweets. A sparse modelling approach is exploited which uses global, regional and user dependant topics and terms distribution in order to geo-reference topics on areas. Resources detected from geo-localised Twitter messages are also utilized to infer transient representation of volatile events happening at venues in [20]. Foursquare places categories are used to create footprints of areas and users in [109] by means of spectral clustering. At the other hand, as regard to more focused works on urban computing, in [25] checkins are used to understand mobility patterns and how these are influenced by users' social status, sentiment and geographic constraints. In the Livehoods project Cranshaw et al. [30] cluster Foursquare venues using spatial and social proximity introducing a new user-based 'bag-of-chekins' similarity algorithm. Although their approach is effective in capturing the social dynamics of cities according to people movements, it is completely lacking

[1] This Chapter has been previously published as "User Profiling for Urban Computing: Enriching Social Network Trace Data" in *Proceedings of the 3rd ACM Multimedia Workshop on Geotagging and Its Applications in Multimedia, 2014* [53].

in considering who those people are and which are their motivations.

The key idea we propose in LiveCities instead is that city venues are characterisable both by static features, i.e. categories assigned by LBSNs on the basis of their type of service, and by dynamic features, i.e. the distribution of the interests of the people who checked-in there, which can change over time. To accomplish this we extract users' profiles of interests and users' geo-localised media automatically from Facebook, then we categorise detected venues using Foursquare APIs and, finally, we weights these features on the basis of semantic similarities and interests distribution. The main contribution of the work is to present our clustering module for city areas identification and classification based on our features selection approach and to show the web application developed for clusters visualisation and venues recommendation.

2.2 The System

2.2.1 Dataset

Through a Facebook app we have collected and gained access to 8839 user profiles, from which we extracted 124790 checkins and identified 52767 venues. Location information is available on Facebook from 2010. Facebook Places started out as a mobile application for people to check into business locations, then it was integrated in Facebook featuring a location tagging tool. People on Facebook can tag specific locations in status updates, image posts, or video posts. Others members can also tag their Facebook friends in specific locations within their updates and posts. Since the most part of the people registered in the application is resident of Florence and its surroundings we chose to conduct our evaluation on this city. The data used for the tests consists of 24031 check-ins and 5321 venues in Florence. Considered that Florence population counted 366443 in January 2013[2] this is a large amount of information. Places were identified in updates, post and events in which the users participated and photographs they were tagged in. Each place has been categorised using the Foursquare API to assign a static label representing the venue's macro-category. As for profiling, users' interests were extracted by retrieving the categories of Facebook pages for which users expressed a 'like'. There are total 398884 'likes' distributed in 216 Facebook

[2] http://demo.istat.it/bilmens2013gen/index.html. Istat data, January 2013

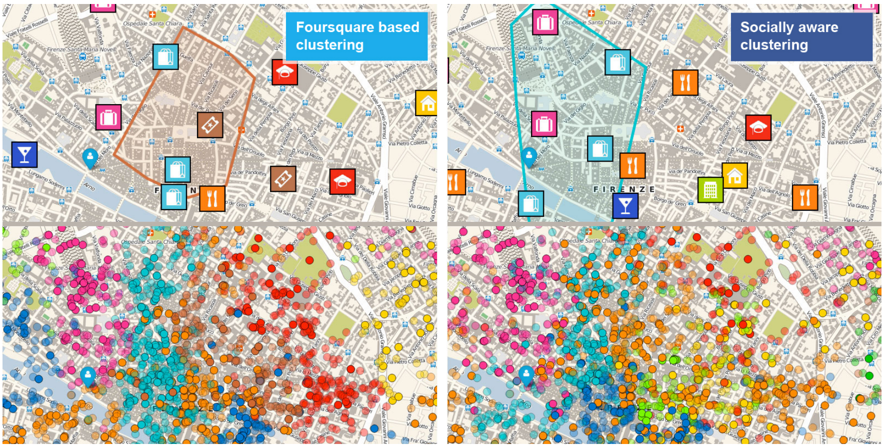


Figure 2.1: LiveCities clusters visualisation of Florence, IT. The figure shows a comparison between the clustering visualisation based on Foursquare categories and the results of our methodology that considers people interest distribution (Socially aware clustering).

categories. User's data is the main reason for which we chose the Facebook APIs to build our dataset instead of the Foursquare or Twitter APIs, commonly used by works in the field [118] [20] [65] [30] [25]. In this respect we can say that Facebook offers, in addition to check-ins data, a higher degree of contextual awareness and an 'environment' exploitable to enrich check-ins semantics.

2.2.2 Clustering module

LiveCities uses k-means clustering to partition the venues dataset into k groups. We run the algorithm on the features selected on the basis of the main idea of this work that people semantics and semantic distances can be exploited to refine places categorisation. Clustering was performed for each city with similarity distances based on different features:

- **Geographic**: latitude and longitude;
- **Foursquare based**: latitude, longitude, Foursquare venue's category;
- **Socially aware**: latitude, longitude, Foursquare venue's category, a weighted vector of interests of the users who checked-in.

These three modalities of features selection have been essential in order to conduct the evaluation and to measure the improvements of our approach (i.e. socially aware). One of the very first problem we have to tackle in our data is that Facebook ‘likes’ categories show an unbalanced distribution. The reason is that some interests like “music” or “sport” are more commonly shared between users than others and that Facebook pages in these categories are more widespread.

To solve this issue, we calculate the weight of a category of ‘likes’ on a venue considering three factors: 1) percentage of ‘likes’ in each category for all the people who checked-in, 2) probability of a generic ‘like’ to belong to a category, 3) semantic distance between each ‘likes’ category and the assigned Foursquare category. Formally, supposing we have a vector F of i_F Facebook places and also a set of L users’ ‘likes’ for each venue, denoting as c a ‘likes’ category, we can compute the weight w for each $c \in i_F$ as follows:

$$\mathrm{w}(c, i_F) = \mathrm{percentage}(c, i_F) \cdot \log_{10}\left(\frac{10}{P(c)}\right) \cdot \mathrm{correlation}(c, i_F)$$

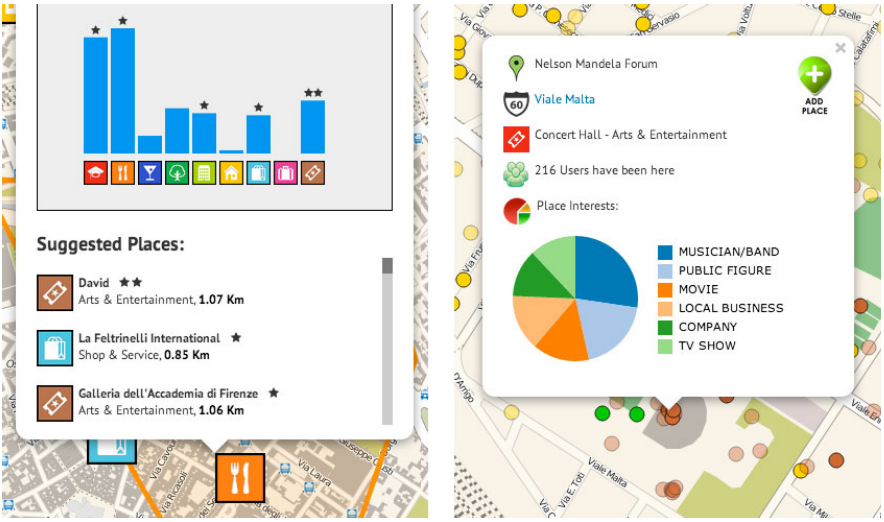
The function uses *de facto* a TF-IDF approach. With $P(c)$ we mean the probability in 2) calculated and normalised on the basis of the distribution of the category ‘likes’ in all the dataset ‘likes’. The correlation function instead uses a semantic distance to compute the affinity between ‘likes’ categories and the Foursquare venues. Distances are pre-calculated and obtained using the Wikipedia Link-based Measure (WLM) by Milne et al. [156]. WLM is a measure for the estimation of the semantic relatedness of two Wikipedia articles through the comparison of their links. In our dataset there is a total number of 216 Facebook categories for pages and 397 types of Foursquare venues, this means that it was necessary to calculate 85752 correlations. To accomplish this, every resource (Facebook category or venue type) has been associated to a corresponding Wikipedia article. We experimented two approaches: 1) manual association, 2) using the MediaWiki API to retrieve possible articles’ matching titles and filtering the results using Latent Semantic Analysis (LSA). Both gave almost the same accuracy. There are two version of the WLM algorithm, the first considers in-bound links and is modeled after the Normalized Google Distance, and the other uses out-bounds links and is defined by the angle between the vectors of the links found within the two articles calculated with the cosine similarity. In LiveCities we re-implemented the algorithm in the latter version because less computationally expensive. To improve the correlation measure, we also observed that when two resources have an high semantic relatedness, often one of the two article contains a link to the other. When this condition occurs, we add a *bonus* to the correlation value.

2.2.3 User interface, personalization and recommendation

LiveCities features a web application based on the principles behind visual analytics for dynamically exploring time-varying, localised and multivariate attribute data relative to city venues and venues customers. LiveCities provides a map based interface and exposes advanced visual components intended to maximise 1) explorative data analysis and 2) service targeting and personalisation.

The application provides two main views, a search view and a clusters view. The search view has been designed as a traditional geographic search interface for venues and it allows users to efficiently filter data by categories or by people interests on the map. The cluster view instead visualises the results of the k-means algorithm. There are three types of visualisation on the basis of three different features selections: 1) geographic, 2) Foursquare-based, 3) socially aware (our approach which takes into account people interests and semantic distances), cfr. Fig. 5.1. Clusters can be visualised as typed squared icons or as set of points. The squared based visualisation uses icons as representative of the ‘centers of mass’ of the detected clusters and allows a less bulky visual access to the information, whilst the points based view show on the map all the venues in the dataset.

Clusters are characterised by different colors, each one corresponding to 9 general Foursquare categories. Points transparency is directly proportional to the computed semantic affinity of the venue category to the cluster classification. In this way colour information is exploited in order to effectively depict points distribution *per* cluster. Clusters boundaries are visualized on user interaction hovering with the mouse over the map, and are calculated using the convex hull algorithm. Users can have statistic insights on clusters and venues through an interactive tooltip, cfr. Fig. 2.2. In particular cluster’s insights present the histogram of venues categories in the cluster and, for each column, the actual geo-referenced venue’s place. Venue’s insights show the distribution of interests of people who checked-in and provide address details and routing. Stars (from 1 to 3) on columns and venues represent recommended resources. LiveCities provides Facebook Login and it profiles



Suggested Places:

David ★★
Arts & Entertainment, **1.07 Km**

La Feltrinelli International ★
Shop & Service, **0.85 Km**

Galleria dell'Accademia di Firenze ★
Arts & Entertainment, **1.06 Km**

Nelson Mandela Forum

Viale Malta

Concert Hall - Arts & Entertainment

216 Users have been here

Place Interests:

Figure 2.2: 1) Insights of a cluster, showing the histogram of venues categories and 2) the distribution of people interests on a venue.

users evaluating their Facebook ‘likes’ on pages, obtained with the Facebook APIs. Recommendation of areas and venues in LiveCities tries to maximise an objective function

$$\max_{p \in places} \mathrm{f}(p, logged\_user)$$

The f estimates the correlation between the user profile of interests and the characteristics of city areas and venues. The semantic relatedness is computed using the WLM measure and weighting suggestions on the basis of users affinity with area’s categories and individual venues.

2.3 Results and evaluation

A preliminary estimation of the results has been conducted for the city of Florence comparing outputs from the three different clustering procedures. We created an online questionnaire with the intent of receiving feedback from city residents about how they perceive the different areas of the city. The questionnaire shows users a map of the city, divided into 15 numbered cells. For each cell, we asked the users to assign labels, according to their mental maps, selecting up to three different categories among those used by LiveCities. We collected answers from 28 users, among 20 and 56 years old and for the most part affirming to have sufficient, good or excellent knowledge of the city (only 4% of the interviewed declared to have an insufficient knowledge). Since clusters shapes are irregular, a single cell can comprehend one or more clusters. On this basis we evaluate how interviewed people labeling of city areas aligns with detected clusters measuring the displacement in the weights of its venues categories. Let A_n be the area of predefined cells adopted in the questionnaire, with $n \in [1, 15]$, we consider the set of clusters OC_n that have some overlapped area with A_n. Formally, for each geographical cluster C_i with $i \in [1, K]$, where K is the number of output clusters of k-means algorithm, $C_i \in OC_n$ only if $A_n \cap C_i \neq \varnothing$. Clusters are described with a multi-dimensional vector formed by weights w_{cat} for every category of the system, with $0 \leq w_{cat} \leq 1$. We define the vector that describe OC_n by computing mean values of the clusters contained in OC_n. We use the data obtained by the questionnaire, represented as a vector of categories weights for every area A_n, as testing data. We can so calculate the Mean Squared Error (MSE) between the expected values (weights in A_n) and the predicted values (weights in OC_n). As an example, figure 2.3 shows intra-categories MSE of each of the three clustering methods for the cell A_{14}. We repeat those steps for every n in order to obtain a global MSE of every clustering method (i.e. geographical, foursquare based and socially aware). The results are the following:

MSE_{geo}	0.059
$MSE_{foursquare}$	0.062
MSE_{social}	0.046

Results show that the MSE in the socially aware clustering approach is lower than with the other ones. Even if the conducted study is still preliminary, results may suggest that our method tend to reflect more correctly the perception that inhabitants have about the characteristics of city areas.

2.4 Conclusions

LiveCities is a web application designed to provide users with a dynamic view of the social patterns characterising city areas and to facilitate resident and visitors in finding places and zones likely to be of interest. Urban

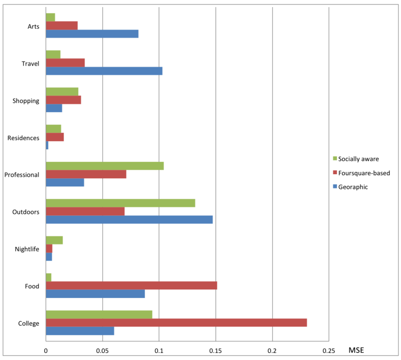


Figure 2.3: Comparison of the MSE in every category for each clustering approach in a case study area of the city.

computation can have a lot of applications, from marketing to trade area analysis, buildings design, urban planning, demographics, entertainments, or simply citizens' life practice. LiveCities offers pictorial depictions of cities and exploits information visualisation techniques in order to shed new light on cities inner workings and on the relationship between people and the environments which they inhabit. In turn it can help to reveal the real 'fabric' cities are woven out. In this demo we showed our methodology for features selection and clustering. We use k-means in order to group venues on the basis both of topological and sociological features. With sociological features we mean that venues are somehow representable not only by their static category assigned by LBSNs but also by the 'bag-of-interests' of the people who checked-in. We also presented the web interface as well as the recommendation and personalisation module. Finally we conducted a preliminary evaluation through an online questionnaire. Results are encouraging and show that our approach deserves to be deepened and that LiveCities can be an useful web tool to suggest to users how to enjoy the best of the places in which they live.

Chapter 3

Separating the Wheat from the Chaff: Events Detection in Twitter Data

This Chapter presents a system for the detection and validation of macro and micro-events in cities (e.g. concerts, business meetings, car accidents) through the analysis of geolocalized messages from Twitter. A simple but effective method is proposed for unknown event detection designed to alleviate computational issues in traditional approaches. The method is exploited by a web interface that in addition to visualizing the results of the automatic computation exposes interactive tools to inspect, validate the data and refine the processing pipeline. Researchers can exploit the web application for the rapid creation of macro and micro-events datasets of geolocalized messages currently unavailable and needed to perform accurate supervised events classification on Twitter. The system has been evaluated in terms of precision. [1]

[1] This Chapter has been previously published as "Separating the Wheat from the Chaff: Events Detection in Twitter Data" in *Proceedings of the Content-based Multimedia Indexing International Workshop (CBMI 2017)* [47].

3.1 Introduction

Social networking sites such as Twitter have become platforms where people communicate and share knowledge on a daily basis through text messages, photos, videos. This huge amount of social data produced by participatory and crowd-generated sensing represent an opportunity to extract knowledge and give valuable insights on urban dynamics. The flow of information from Twitter is realtime, covers diverse topics and can describe actual events. These events can even be quite small such as parties, companies' presentations, road accidents and so on. Well-established networking mechanisms can improve the information gain through the analysis of this flow's dynamics. Number of messages in time, social reactions such as *likes*, *retweets* and direct replies, geolocation along with multimedia features can be processed in order to detect the occurrences of events using statistical models. In November, 2016 Twitter counted 317 active millions users. Roughly 80% of these users access the platform through a mobile device and about 2% of them choose to share their GPS location. This is a small percentage but nevertheless it is a large number of people who produce significant contextual information, especially in densely populated urban areas.
In this Chapter a method for unknown event detection is proposed which relies on this geolocated data. The method uses a lightweight statistical approach and can alleviate common issues on this subject related to computational cost and systems scalability. Most of the works in the literature exploit content-based unsupervised approaches for event detection with a considerable computational complexity (see Sec.3.2). The huge amount of text streams to be processed in realtime, the sparsity of geolocalized data and the noisy nature of Twitter messages are some of the main obstacles researchers have to cope with. These issues make often unfeasible an implementation in a real system.

In this Chapter we describe a method based on geographic spatial grids and implemented in a processing pipeline, explained in Sec. 3.3, which combines several algorithms for statistical analysis without exploiting mining techniques highly computationally intensive. The approach focuses on the analysis of temporal and spatial characteristics of tweets distributions in order to detect abnormalities and accumulations. The method can be effective for macro and micro-events detection as it accounts for the historical time series in terms of volumes and density of data. Supervised approaches can of course perform better but they may require an hard and time consuming work for tweets discovery and labeling not feasible for unspecified events. In order to support supervise classification, our method has been implemented in a web interface which provides researchers with tools to validate and categorize events automatically discovered by the system and to store them in datasets of geolocated tweets. To the best of our knowledge, there are no datasets of Twitter macro and micro-events available to researchers which contain exclusively geolocated data and cover diverse topics. Sec. 3.5 reports the result of an evaluation of our system in building a dataset of events with geolocated messages published in London and New York.

3.2 Related Work

Previous studies have addressed the problem of detecting events in social data and specifically on Twitter through the identification of abnormalities in its temporal flow. Exploited features are mainly frequency and density of terms, hashtags, named entities, reactions, emoticons. All these works use a variety of techniques ranging from K-means clustering, SVM, gradient boosted decision trees to generative language models and temporal query expansion [10, 93, 95, 116, 126].

Wang et al. [152] improve clustering quality enriching Twitter messages with term expansion on *Wordnet*. Generative models and statistical clustering on textual data from Twitter has been used more recently in [151] and [83]. Nguyen et al. [106] consider keyword occurrences (Occurence-Score) over time, number of participants involved (Diffusion-Degree) and speed of information spread (Diffusion-Sensitivity) to calculate the probability that an event occurred on the basis of term score distribution. Hierarchical clustering of terms and *Wordnet* expansion have been used in [145] for emerging event detection. The authors observe changes in events' popularity defined as number of messages in events clusters. Irregularities in the rate of messages are exploited in [119] and in [59] where also location and topical clustering is performed. As regard to applications, closest to our work is the *CityBeat* [161] system which detects abnormal signals combining time series and classification methods based on spatial, meta, textual and historical features. The main issue with existing approaches on event detection from tweets is mainly the computational cost of extracting and elaborating a lot of features. Twitter messages are composed by very small sentences filled with mispelled words, hashtags, symbols, urls that need to be cleaned and normalized. This noisy realtime data is huge and not easily manageable from a computational point of view. Studies such as [83, 106, 151, 152] aim to function in real-time on all this data but their performance is poor in terms of processing time and system scalability. Furthermore performances are evaluated only on small *corpora* or not evaluated at all.

For addressing these challenges, we propose a processing pipeline designed keeping in mind that an event occurs in time and space. From our perspective, in order to reduce processing time and implement algorithms in a usable tool it is feasible to reverse common approaches and to take into account exclusively geolocated data. This may be regarded as a limit but we think it is definitely a *plus* for the main objective of our system which is the implementation of a method for the detection of geolocalized macro and micro-events exploited by a tool for events datasets creation. In fact, this choice 1) decreases false positives; 2) filters the information reducing the amount of data to be processed; 3) allows the rapid creation of datasets of events through our tool, that otherwise would require a considerable amount of work for searching and inspecting the Twitter knowledge-base.

Our pipeline combines some algorithms and techniques, explained in Sec. 3.3 and is implemented in a web system. The system provides an exploratory interface which allows to refine and improve the detection adjusting the spatial and temporal parameters exploited by the algorithms, allowing a fine-grained analysis which works on subsets of data. The main outcome of this work is the design and implementation of a lightweight and configurable system for *1)* the semi-automatic detection of macro and micro-events of urban geodata extracted from Twitter (see Sec. 3.3) and *2)* the easily creation and management of datasets of micro-events, currently unavailable for research studies on Twitter data[2].

3.3 Detection and Mining

The proposed processing pipeline used in our system contemplates three main steps: *a)* tweets extraction; *b)* abnormalities detection; *c)* mining and visualization.

[2] video available at `https://vimeo.com/miccunifi/twitter-events-detection`

Tweets extraction tweets are extracted daily through the Twitter API using the Java library twitter4j [3] and stored in a mySQL database for post-processing.

Abnormalities detection we use a grid-based method to analyze all the geo-referenced data in a certain urban area. The area is divided arbitrarily in cells of predefined dimensions. Tweets in each cell of the grid are analyzed with our method which combines temporal and spatial density analysis. The idea of the method is to firstly divide tweets spatially into a grid, then to use time series analysis (exploiting DTW and Crest Detection) to detect anomalies in the volume of tweets of the cell, and finally exploit spatial clustering to infer a possible event from anomalies. For each cell c, we create a time series V containing the number of unique users per hour who have published at least a geolocated tweet in the cell c. We define a time interval T (e.g. 24 hours), and we divide the initial time series V in windows of size T obtaining a set of time series V_i, with $i \in [0, N_{windows}]$. We then compute the average time series $\overline{V}$. On this basis we perform our method for abnormalities and event detection which consists in a pipeline of three core algorithms described below:

1. **DTW** Dynamic Time Warping is an algorithm which allows to measure the similarity and the distance between two time-series. DTW is widely adopted in information retrieval to cope with deformations of time-dependent data [132]. For each time window i, we compute the DTW distance between the time series V_i and the average time series $\overline{V}$, obtaining a measure of distance d_i. To detect windows of time where the distribution of tweets is unusual, we consider the average of the distance $\overline{d}$ and the standard deviation σ_d. All the time series for which $d_i > \overline{d} + \sigma_d$ or $d_i < \overline{d} + \sigma_d$ are considered as abnormalities and sent to the step 2 (Crest Detection).

2. **Crest-detection** is an algorithm which uses peaks windowing in order to detect anomalies in data distribution. We consider the time series marked as abnormal in step 1. For each time-step t, $t \in [0, T]$, we compare the values of unique tweetters in that hour $v(t)$ with the values of the time series in a temporal interval of size Γ_t that precedes and follows t. A peak is detected at time t if $v(t) > \sum_{j=1,..,\Gamma_t} v(t-j) + \epsilon$ and $v(t) > \sum_{j=1,..,\Gamma_t} v(t+j) + \epsilon$.

[3] `http://twitter4j.org/en`

Values of Γ_t and ϵ are set to 2 as default (customizable through the interface, see Sec. 3.4)

3. **DBSCAN** Density-based spatial clustering of Applications with Noise [40] is a data clustering algorithm that given a set of point in the space groups together the points that are closer in the distribution. The algorithm divides the points (i.e. tweets with *lat, lng*) in 'core points', 'border points' and 'noise points'. A point p is a 'core point' if at least a minimum number $minPts$ of points are comprised in a distance ϵ and are directly reachable from p. Mutually density-connected 'core points' form a cluster. A point q is a'border point' part of the cluster if a path exists between p and q so that all the 'core points' in the cluster are density-reachable from any point of the cluster itself. All the other not reachable points are considered 'noise points'. An event is proposed by the system if at least one cluster is detected by DBSCAN for the tweets posted during the temporal interval of the peaks detected in step 2. As default the system sets $\epsilon = 5$ meters and $minPts = 3$ (customizable using the interface).

Mining and visualization Once the event clusters has been identified by the DBSCAN algorithm, the event is visualized on the web interface and positioned in its center of mass with respect to the geolocalization of each tweet (see Sec. 3.4). Text mining is performed on the tweets that are part of the cluster in order to show content features of the detected event. We extract: *I)* most frequent words; *II)* hashtags; *III)* named entities (i.e. timestamps, names of persons and organizations, places); *IV)* attached photos and videos; *V)* part-of-speech tagging. Visualization of metadata and related multimedia material is essential in order to help users in verifying the correctness of the event detection and to build categorized datasets.

3.4 The Web Interface

The Web Interface has been developed in Java and deployed as a servlet in a Tomcat container [4]. The interface has been designed with the main goal to visualize on a map the results of the automatic event detection pipeline described in Sec. 3.3. Furthermore, the web system exposes semi-automatic tools which may help researchers to tune the exploited algorithms in order

[4] `http://tomcat.apache.org/`

to improve the event detection. Not least, the system allows the creation of datasets of events from Twitter.

The interface provides two main views (see Fig. 3.1), both map-based [5], with two different access levels: *1)* the visualization view shows on the map the events automatically detected by the system; *2)* the validation view allows an authorized user to confirm (or not) the correctness of the detection and to customize the search.

In the visualization view each event is represented by a marker that can have three color: blue, red and green. Blue pins are the events detected by the algorithm but still not validated by a human; red and green pins instead are events respectively misclassified or correctly classified as confirmed by a user. The user can search and zoom the map and define a temporal range for the visualization of events detected by the algorithm with the defaults parameters. Each marker can be activated in order to open an info-box window which shows the data and metadata associated with the event: the hour, the category and all the extracted features (tweets, word occurrences, related multimedia, named entities, POS tagging). Authorized users can access the editing and validation view. The view shows a transparent grid super-imposed on the geographic area of interest. The dimensions of the cells of the grid are predefined and the grid is positioned arbitrarily to cover all the geolocalized tweets published in a configurable radius. The user can select a cell, as shown in Fig. 3.1, in order to show an info-box where all the events detected in that area can be inspected. In this modal interface the user is also provided with advanced graphical widgets through which he can adjust the several parameters used by the algorithms. Hence, the computation can be started again asynchronously in order to discover events previously not detected by the algorithm with the default parameters. Configurable parameters are: *1)* the position and the dimensions of the grid cell (Fig 3.2.1); *2)* the time interval of the event detection; *3)* the time period and/or periodicity over which to calculate the average and the standard deviation of the DTW distances for the cell; *4)* the threshold value over the average *plus* the standard deviation beyond which the system reports an abnormality; *5)* the time window and the ϵ used in the crest-detection algorithm (Fig. 3.3); *6)* the *minPts* and the ϵ used by the DBSCAN algorithm for the identification of clusters of tweets on the peaks detected by the Crest Detection algorithm (Fig. 3.2.2). The flexibility provided by the system in tuning parameters

[5] Interactive maps are provided using `http://leafletjs.com/`

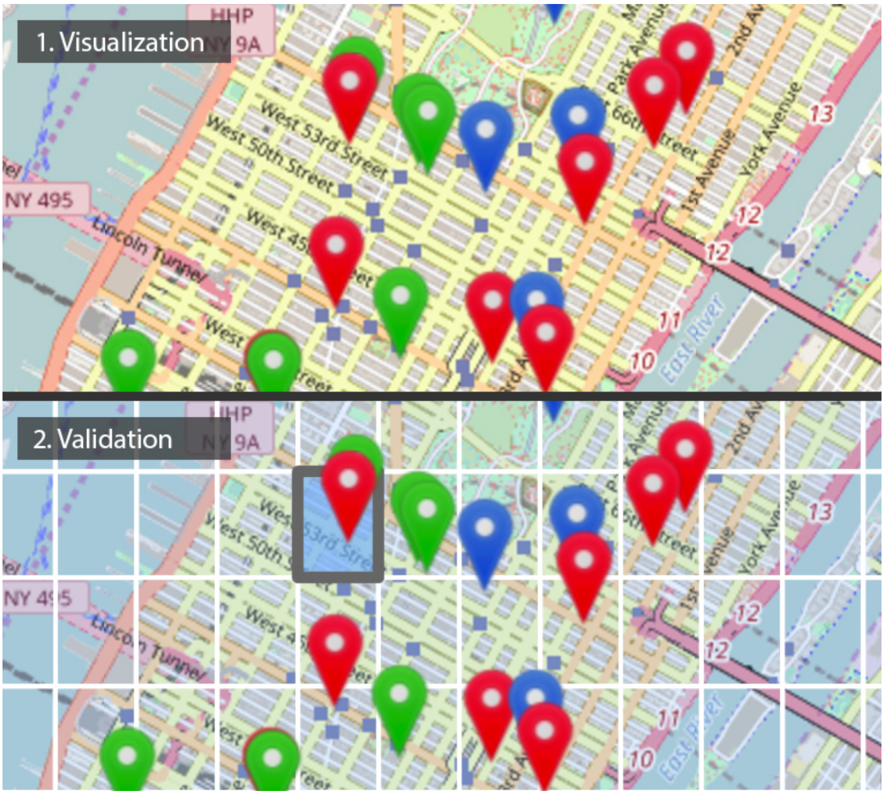


Figure 3.1: The two main views of the system: 1. Visualization. 2. Editing and Validation

as well as the configurability of spatial cells dimensions, density and time intervals allow the system to work in realtime on subsets of data with good performance. Furthermore, the choice between periodicity and/or temporal continuity for the computation of historical data can improve the search. In general big events tend to be periodical (e.g. football games are usually played every two weeks) and an abnormality could be detected comparing, for example, the time series of the day to the time series of the last ten days. On the contrary, a micro-event such as a small company meeting can increase the number of tweets in a certain day with respect to a periodicity average (e.g. every Thursday) but not in a continuous time interval.

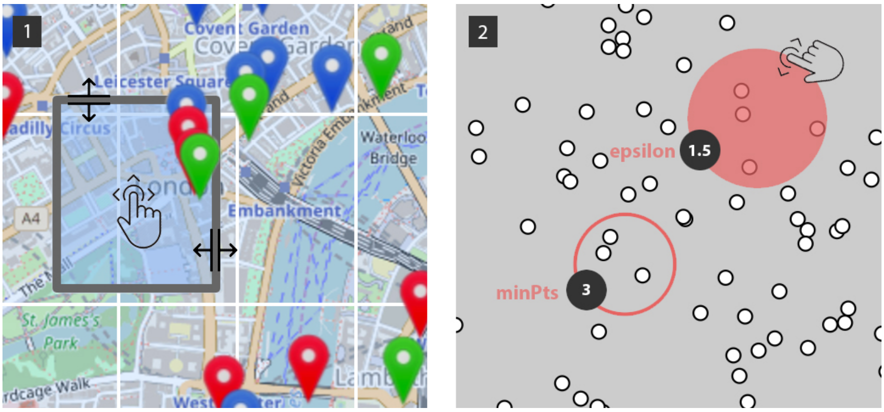


Figure 3.2: In 2.1: the draggable and resizable widget to select a specific area on the map. In 2.2: ϵ and *minPts* of DBSCAN can be defined dragging the circles' circumference on a distribution of points.

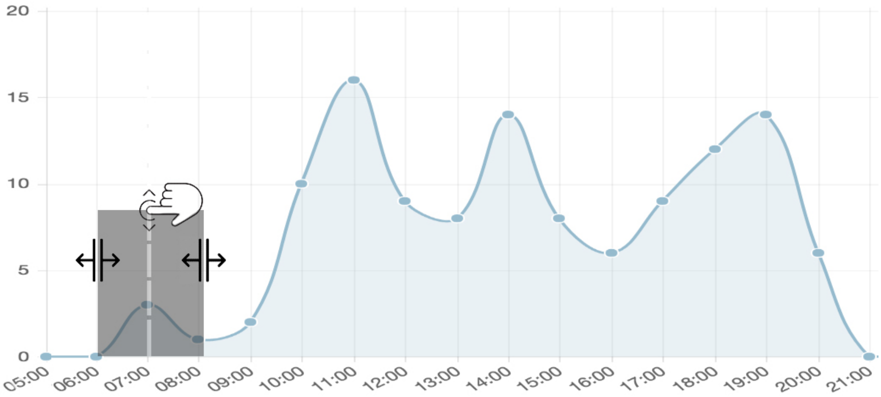


Figure 3.3: Time series of Twitter users who published geolocalized posts in Trafalgar Square, London, on 2016, April 1. The system detected 4 events at 7am, 11am, 2pm and 7 pm. ϵ and time interval can be changed interactively resizing the gray rectangle on the time series plot.

3.5 Evaluation

To evaluate the detection accuracy of our system we manually validated the events automatically discovered between March 31 and April 9, 2016. The tweets published in the city centers of two big cities in these ten days, London and New York, were analyzed. This time period was chosen since the interval registered the highest number of tweets in the year. In details 17176 users published 44932 tweets in London, while 17378 users published 43186 tweets in New York provided with geolocalization. The system detected a total of 1240 events, 340 in London and 900 in New York using the algorithms with default parameters. The significant difference in the events' count is probably due to the diversity in population density in the two city centers (30000/km^2 in Manhattan and between 10.000/km^2 and 15.000/km^2 in the central districts of London). The overall error rate of the classification was 0.43 with 190 confirmed events in London and 516 in New York. This is a very good result in terms of precision, not far from state-of-the-art supervised approaches for event detection which range from 0.64 to 0.85 [59]. Results of a recent method following a more similar approach on geolocated tweets and Instagram photos [119] achieved a precision of just 0.20. The recall of the system has not been computed due to the unavailability of Twitter annotated datasets of macro and micro-events provided with geolocated messages.

3.6 Conclusion

In this Chapter we have presented a lightweight method for the automatic detection of unknown macro and micro-events exploiting geolocalized data from Twitter. The system uses a combination of algorithms to discover possible events using a pure statistical approach. The method is exploited by a web system which helps researchers in building datasets of geolocalized events. Default parameters of the algorithms can be changed on the fly in order to refine the detection and to validate and categorize the events proposed. These data can be useful to other researchers for improving supervised event detection and classification techniques on Twitter messages.

3.7 Acknowledgment

Research partially supported by MIUR Cluster project Social Museum and Smart Tourism.

Chapter 4

Roadie: Mobile Semantic Tourism Routes

Roadie is a mobile application for the planning and the recommendation of tourism routes in cities. Recommendation is achieved profiling users in a semi-automatic way via social network analysis and profile curation, and by monitoring user activity analysing data coming from devices' physical sensors. Routes consist of several POIs (Point Of Interests) that can be automatically or manually enriched with geo-events and geo-services obtained dynamically from the web, categorised and suggested using semantic analysis. [1]

Roadie[2] is a mobile-based application with two main goals: 1) to provide accurate recommendations of city itineraries enriched with thematic suggestions based on user profiling, 2) to let the user manually create and edit his/her personalised tours through the city. The main contribution of the application is to provide a multi-dimensional contextual approach [58] for recommendation. In fact Roadie combines user profiling on social networks, location-awareness, semantic analysis and activity recognition by sensing in order to improve the personalisation and the recommendation of city tours

[1] The work presented in this Chapter has been published as "Roadie: Mobile Semantic Tourism Routes" in *Proc. of IEEE International Conference on Multimedia & Expo (ICME) - Demo Session, 2015* [54].

[2] Video available at http://vimeo.com/miccunifi/roadie or downloadable from http://bit.ly/1o3V1cY

in mobile electronic guide systems.

4.1 The system

Roadie has been developed in Java as a native mobile application using the Android SDK. The backend is written in PHP. All the data are stored in a MySQL database. The system (Fig. 4.1) is composed by four core modules which are respectively in charge of: 1) grabbing POIs, venues and events, 2) categorising data in realtime, 3) profiling users, 4) recommending routes.

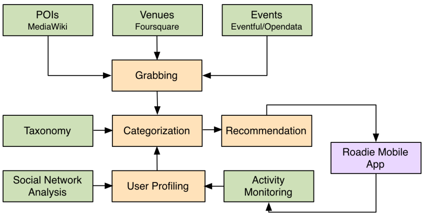


Figure 4.1: Roadie System Architecture.

4.1.1 Data Collection

Roadie uses two types of geo-localised data: static data, which doesn't need to be updated frequently, such as POIs and venues, and dynamic data, mainly constituted by events, which change continuously in time. POIs are city attractions retrieved querying the MediaWiki API[3]. For each POI a textual abstract, a representative image, and place categories (e.g. church, museum, palace, etc.) are collected. GeoNames API[4] is exploited in order to get POI's latitude and longitude. Venues are places in the city which can offer visitors commodities, leisure and entertainment services, such as shops, restaurants, nightclubs etc. These data are obtained using Foursquare API. Roadie uses events to enrich the visitor experience in the city. Events are characterised by having an exact start time and duration and are retrieved daily by the grabbing module from two sources: the OpenData (for this demo we used data published by the municipality of Florence, IT), and the

[3] http://bit.ly/RywgdI

[4] http://www.geonames.org/

Eventful API[5]. The first is an institutional source which provides especially art exhibitions and events, the latter instead is user-generated and concerns, for the most, musical events and shows.

4.1.2 User profiling

Roadie builds a model of the user profile of interests exploiting user information from the Facebook Graph API through Facebook Login. User interests are extracted by analysing the categories of Facebook pages for which the user expressed a 'like'. Additional basic demographic data such as age, gender and residence are also collected. In order to assign profiles to users that are not able or reluctant to connect their Facebook profile, we adopt an inference method based only on the demographic infos. Given the user u, Roadie looks for similar registered users and gets a subset of users S_u. From S_u, the most frequent categories of interest are extracted and assigned to the interest profile of u.

Monitoring user activity

Roadie exploits also physical sensors available in mobile devices in order to observe user behaviours and to refine his/her profile model. We identify two possible meaningful activities for the tourism domain: 1) speed of user's movements between geographic positions is monitored in order to estimate running/jogging activity, 2) altitude peaks are analysed by a probabilistic model to assess a user preference for climbing or panoramic views. An asynchronous thread processes data from sensors even when the mobile application is in background, allowing a continuous monitoring of the user movements. To minimise battery consumption, data are sampled and processed every 60 seconds. Furthermore, the thread is stopped in case of a low battery level (less than 25 %). Activities, when detected, are associated with corresponding categories in the knowledge-base and in user profiles as summarised in table 4.1.

Categorisation of resources and Recommendation

The categorisation module is responsible to classify all the resources coming from the grabbing module and to compute a semantic similarity between

[5] http://opendata.comune.fi.it/, http://api.eventful.com/

Table 4.1: User actives and related detected interest

Activity	Analyzed data	Extracted interest
Running	Speed from GPS data	Sport
Sightseeing	Peaks in Altitude	Lookout

user preferences and resources to be suggested. Roadie adopts a taxonomy of 19 macro-categories that has been manually defined in order to exhaustively represent both user interests and venues/events. Since data are continuously updated from heterogeneous sources (e.g. events from Eventful, user interests from Facebook, venues and places from Foursquare), uncategorised or labeled with different categories, Roadie analyses textual information in order to map these data according to the system's taxonomy. A max similarity score is computed between each item c of the taxonomy and a provided or inferred resource category r using the function $\mathrm{sim}(c, r) \in [0, 1]$. The correlation is estimated using a semantic text similarity technique [63]. The method is based on distributional similarity and Latent Semantic Analysis (LSA), further complemented and improved with semantic relations extracted from WordNet[6]. Recommendation of routes is based on several factors: context inference, profile of interest computed analysing social network data, user activities detected from device sensors. Given the user location, the recommended routes are built through the MapQuest Route Service[7] trying to maximise the ratio between the number of attractions to visit and the available time. Venues, places and events with the highest semantic similarity to the user profile of interests are categorised and suggested, ordered by distance from the user.

4.2 The application

Roadie is a native mobile application developed in java using the Android SDK. The interface is composed by four main views: 1) recommended routes, 2) route creation, 3) profile and 4) saved routes. Roadie features Facebook Login as well as app registration. The profile view shows demographic data and all the automatically detected interests. These can be edited adding or removing items from the system taxonomy. The recommended routes

[6] https://wordnet.princeton.edu/

[7] http://mapq.st/1jAVgeL

view provides a scrollable list by which the user can select different thematic itineraries on the basis of his/her interests. For example, if the system detected sport and literature as the main user interests, the recommended route is enriched with sport or public readings events going on at the moment in the city. All the routes are presented on interactive maps provided by the OpenStreetMap[8] service exploiting MapQuest web mapping features. Recommended routes can be edited manually directly on the map or users can utilise the route creation view to plan their visit from scratch. To this end Roadie provides smart suggestions mechanisms which offers an unified perspective for tourist attractions and contextual services. First of all, it proposes POIs taking into account user position. Each time the user selects a POI the system suggests other possible POIs nearby. Otherwise he/she can search for one. Once the route has been defined, it can be enriched with venues and events relevant with user interests and POIs he/she has planned to visit. Imagine a use case scenario where a user is building a route that will take him from the Florence Cathedral through the Uffizi Gallery and then to Piazzale Michelangelo in Florence, IT. If the system detected that sport and food are among user interests, he/she will be suggested the nearest restaurant to the Uffizi Gallery for lunch or to take part in a running race scheduled for that day and starting from Piazzale Michelangelo. The sport interest may have been manually added by the user, inferred from Facebook data analysis or detected by Roadie monitoring data from the user smartphone sensors, assuming that the user is a regular jogger.

[8] http://www.openstreetmap.org/

Chapter 5

Exploring 3D Virtual Environments through Optimised Spherical Panorama Navigation

Commonly, immersive and virtual reality systems simulate real environments exploiting 3D computer graphics. This entails a considerable work to be done in models development and objects textures mapping in order to obtain a good degree of realism. Furthermore, the excessive complexity and the high rendering quality of the models can compromise system performance, especially in a web environment. This paper describes a vision based approach which allows a user to immersively navigate a real cultural environment through a lightweight web based system for the interactive walk-through and browsing of an ordered sequence of spherical panoramas. [1]

[1] The work presented in this Chapter has been published as "Exploring 3D Virtual Environments through Optimised Spherical Panorama Navigation" in *Proc. of IEEE International Conference on Multimedia and Expo (ICME 2015) - Demo Session, 2015* [46].

5.1 Introduction

In the last few years web browsers have been providing an increasing support to third dimension technologies, although 3D rendering engines are not fully ubiquitous and there are still some issues regarding the performance and realism of these systems. Many researchers have then focused on panoramic images due to their attractive and immersive display usage, first of all focusing on how to build a panorama starting from image sequences (two or more) [27], and then addressing the panorama navigation issues. Several works exist which use different mapping representations for outdoor panorama navigation, for example Google Street View [4]. However the navigation metaphor is often more intended to give users a good experience of exploring one panorama (panning, zooming and titling), rather than to optimise the switching from one panorama to another. Our goal is to enable smooth transitions across panoramas so as to reduce their perceived discontinuity and to propose an innovative interaction metaphor for a better multimedia fruition including 3D models, PDFs, galleries of images/videos and indoor panoramas related to the Point Of Interests (POIs) distributed along a cultural walkthrough.

The paper is organised as follows. We discuss the image-based method for panoramas transition in Section 2. Section 3 presents the web interface and its interaction design metaphor. Section 4 outlines conclusions and future work.

5.2 Optimised spherical panorama navigation

A spherical panoramic image is created by warping the radially undistorted perspective images onto a unit sphere assuming one virtual optical center in which the user is supposed to watch at the panorama. Navigation within a single panorama is provided by two main actions: dragging and zooming; navigation among different panoramas is performed through a smooth transition with the replacement of the panoramic texture. In order to support the navigation between panoramas the interaction design model uses a metaphor based on the zooming action of the user: the transition takes place only when the current zoom level exceeds a particular threshold (see Sec. 5.2.1). In order to minimise the gap in the transition between the two scenes, we studied and developed an algorithm that performs sub-scene matching.

5.2.1 Sub-scene matching algorithm

The sub-scene matching algorithm is exploited in order to find the best zoom factor threshold for triggering the transition to the next (or previous) panorama. Given two images I_k and I_{k+1} we want to find the rectangular sub-region of I_k whose content is most similar to I_{k+1}. The sub-region and I_{k+1} must have the same aspect ratio, although their size is expected to be different. Let us introduce the following notation:

- w_k and h_k are the width and height in pixels of the image I_k;
- the function $Crop(I, x, \Delta x, y, \Delta y)$ returns the rectangular sub-region of the image I identified by the bottom-left coordinates (x, y) and the top-right coordinates $(x + \Delta x, y + \Delta y)$;
- the function $D(I_n, I_m)$ returns the dissimilarity between the content of two images I_n and I_m : zero in the case that the content of the two images is the same and a number greater than zero otherwise. The measure of dissimilarity is invariant to the size of the two images.

Identification of the rectangular sub-region of I_k that best matches the content of I_{k+1} in correspondence to the inward direction is accomplished by minimisation of the following cost function with respect to the three variables (x, y, η):

$$\begin{cases} (x_0, y_0, \eta_0) = arg[\min_{x,y,\eta} D(f(x, y, \eta), I_{k+1})] \\ f(x, y, \eta) = Crop(I_k, x, x + \eta w_{k+1}, y, y + \eta h_{k+1}) \end{cases} \tag{5.1}$$

The result of Eq.(5.1) is the triplet (x_o, y_o, η_o) of the variables that minimise the cost function; they are the coordinates of the bottom-left vertex of the sub-region $P_o = (x_o, y_o)$ and the zoom factor $Z_o = 1/\eta_o$ to be applied to the sub-region of I_k in order to match the size of I_{k+1}. Experimentally, we observed that computation of the dissimilarity function $D(I_n, I_m)$ through the distance of the image histograms provides higher effectiveness compared to solutions based on scale invariant local keypoint descriptors such as SIFT. This is mainly caused by the fact that in the general case, the scenes represented in two consecutive panoramic images can differ significantly in some parts, due to severe occlusions that can take place depending on the 3D structure of the scene captured by the panoramic images. To reduce the computation time associated with the minimisation of the cost function and speed up the computation of the histograms we adopted the technique of Integral Histograms [117].

5.3 Application Interface

The system has been developed as a web application based on HTML5, CSS3, Javascript and WebGL and has been implemented using the 3D ‘open source’ library *three.js*[2]. The navigation among spherical images has been carried out through the dynamic replacement of the texture mapped into the sphere, according to user interaction. The update of the texture is done taking into account the optimal zoom factor threshold for the transition (see Sec. 5.2.1). This can occur when the user zooms in the outward direction and exceeds the threshold pre-computed by the sub-scene matching module. Otherwise, when the direction of zooming is not aligned to the outward direction the rendered image is progressively magnified—and the field of view is reduced accordingly—until the maximum zoom level is reached. The application can be configured in order to show interactive hotspots in the virtual walkthrough. Each hotspot can be activated by users and it provides several multimedia additional material about specific POIs. Hotspots are constituted by four graphic icons arranged in a circular menu. Each item represents one of the following multimedia categories: 1) gallery of images/videos (yellow icon), 2) PDF (red icon), 3) 3D object (green icon) , 4) Indoor panorama (blue icon), cfr. Fig. 5.1. All these multimedia artifacts are shown inside a floating panel oriented contextually to the panoramic environment and are arranged in real-time by the system with the best spatial position according to the user point-of-view. In this way, the application overcomes the limitation of standard panorama-based interfaces in which additional content (3D or 2D) is shown in two-dimensional *lightboxes* covering the main navigation area. Furthermore, hotspots can be dragged using an handle at the centre in order to let users better organise the content in the interface view. Context awareness is provided through a mini-map at the bottom-left angle of the screen. Position and orientation are shown on the map and a little circular handle on the route path can be dragged in order to move to different panoramas. POIs, hotspots and associated multimedia materials are searchable using an autosuggest input field.

[2] http://threejs.org/

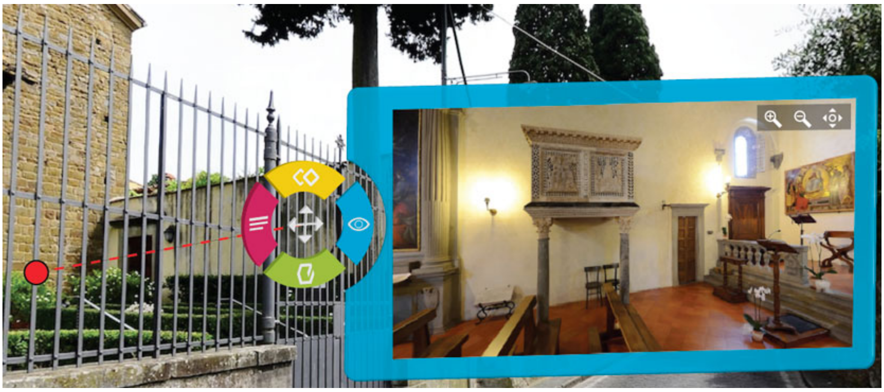


Figure 5.1: Examples of interaction with an hot-spot: Indoor panorama

5.4 Conclusion

In this paper we present a web tool for an immersive interactive walk-through in an urban cultural scenario and we propose a new interface and interaction metaphor for accessing cultural content. An optimisation method for the transition between spherical images based on a Sub-scene matching paradigm is proposed.

5.5 Acknowledgments

The research was conducted as part of a project developed in collaboration with F.A.S.T S.R.L. (Fine Art Science and Technology), which provided all the multimedia material, and funded by Regione Toscana.

Chapter 6

Item-Based Video Recommendation: an Hybrid Approach considering Human Factors

In this Chapter we propose a method for video recommendation in Social Networks based on crowdsourced and automatic video annotations of salient frames. We show how two human factors, users' self-expression in user profiles and perception of visual saliency in videos, can be exploited in order to stimulate annotations and to obtain an efficient representation of video content features. Results are assessed through experiments conducted on a prototype of social network for video sharing. Several baseline approaches are evaluated and we show how the proposed method improves over them. [1].

[1] A preliminary version of the work presented in this Chapter has been published as "A system for video recommendation using visual saliency, crowdsourced and automatic annotations" in *Proceedings of the 23rd ACM international conference on Multimedia, 2015* [50] and then as "Item-Based Video Recommendation: An Hybrid Approach considering Human Factors" in *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval, 2016* [49]

6.1 Introduction and Related Work

Collaborative Filtering (CF) is a technique often used by Recommender Systems (RSs) which aims at predicting interesting items to a user based on the preferences, explicit and implicit, of other users. A standard item-based video RS builds its prediction model considering user preferences for videos, expressed according to ratings, and suggests potential videos of interests comparing their distributions. Hybrid approaches in RSs have been proved to give best results [44]. These approaches combine CF with content-based techniques and reduce issues related to the large amount of data to be annotated and data sparsity. Recommending relevant videos can help users to find the most pertinent content according to their view habits or preferences. As shown in [171], recommendation is a powerful force in driving users to watch other videos, much more than direct search of new videos. Hybrid approaches presented in the literature typically exploit textual video metadata, sometimes complemented by multimedia content analysis [165], user profiling, social features and User-Generated Content (UGC) [11, 32, 89]. Crowdsourced data is usable information that can be leveraged to improve different online services.

In [29] crowdsourced annotations are used to create video previews that are more related to the queries of the users, to improve video retrieval. In [135] the performance of a video retrieval system based on crowdsourced annotations of sport videos shows that despite the heterogeneity and poor quality of the annotations, they are close to ground-truth.

Time accurate annotations of social videos, based on user comments and temporal, personalised topic modelling, has been proposed in [159]. In [163] a large crowdsourcing experiment has been carried out to analyse the differences between “timed” tags (i.e. added to a specific timecode in a video) *versus* “timeless” tags. The authors observed that most of the visually-related tags are relevant for short segments of the video, i.e. people tend to tag when something is “flashed” in the video. We build on top of these studies and propose the adoption of an hybrid approach in which a brief and comprehensive representation of video content can improve the performance of a standard recommender based on CF (i.e. using only ratings). The approach relies on content-based features gathered both through crowdsourced and CNN-based classifiers annotations. The dataset has been collected through a prototype of a Social Network (SN). Annotations collection is improved exploiting two human factors: *i)* user profile interfaces and *ii)* video frames visual saliency.

The main goals are: *i)* to increase the number of crowdsourced annotations, that provide an enrichment of automatic video annotations; *ii)* to improve the quality of video recommenders through video content analysis.

The Chapter is organised as follows: in Sect. 12.2 the social network architecture and modules are described. Experimental results are presented in Sect. 11.4 to show the influence of user profiles and visual saliency on the collection of user annotations of videos. Evidence is given that systems featuring a user profile interface stimulates user activity, increasing the number of annotations. We also show that frames with an high visual saliency are more likely to be annotated; this can be used as a criterion *i)* to suggest to users relevant frames; *ii)* to filter relevant frames for automatic annotation. The recommender is evaluated in Sect. 6.3.3.

6.2 The system

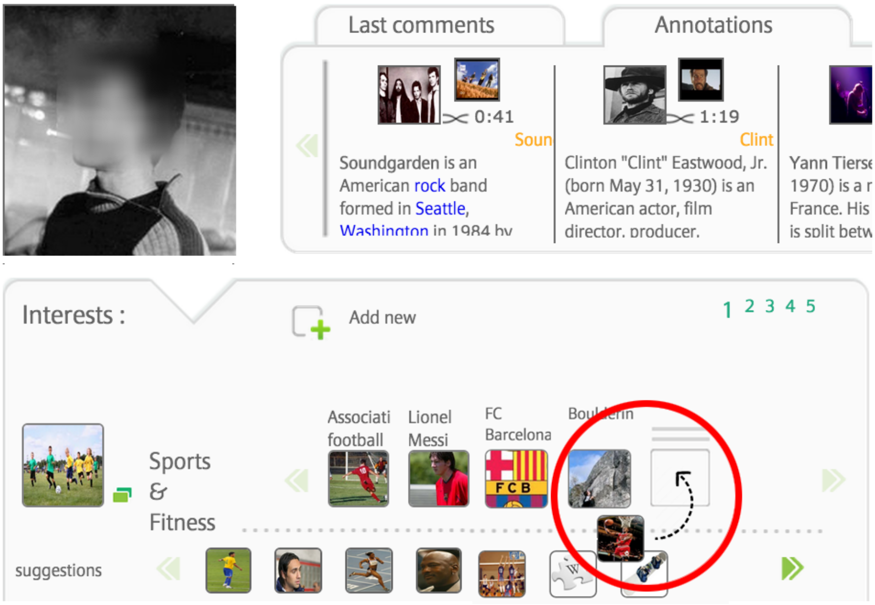
The item-based RS has been implemented in a prototype of a SN[2]. The idea behind the SN is to exploit user profiling techniques to propose to the user targeted recommendations of videos, exploiting suggestions of topics of interest and similar users. This is achieved tracking user's activities on the SN, such as comments, number of video views, click-through data and video ratings. Users can comment videos at frame level tagging concepts derived from Wikipedia. All the concepts manually added are clustered in 54 categories using Fuzzy K-Means and classified using a semantic distance [97] with a kNN approach. Categorised resources in videos are used to build a vector describing video content, then exploited in the RS. The SN also allows users to build a public personal profile of resources of interest from those extracted from comments or added by the SN users. The profile module is exposed relying on the hypothesis that self-expression and *self-esteem* can be exploited to engage the user in the annotation process, in this way easing the collection of crowdsourced annotations. Salient frames of each video are extracted and related users activity on them is monitored in order to verify if visual saliency can affect user engagement with the system. The positive correlation, verified in Sec. 6.3.2, is exploited at the interface level for easing the annotation process proposing a widget of most salient frames above each video. Automatic video annotations are then extracted using a CNN-classifier on the more salient frames.

[2] `http://fiona.micc.unifi.it/intime`

User profile interface As noted in [169], profile curation is inherent to the use of SNs since management of personal content is integrated with its generation. The content that people choose to share online has to do with how they curate their self-image and present themselves to others. In a 2013 survey, participants ranked their relational identities as most important to them when sharing content on social media [100]. SNs such as Facebook and LinkedIn, for example, are commonly regarded as a space for personal self-expression and self-promotion [147]: users shape their identities in order to gain popularity and reach more and more recognition and connectedness. Our prototype system provides users with a public profile that can be curated in a semiautomatic way. The profile shows user's last comments and annotations as well as annotated video frames and tagged Wikipedia resources with thumbnails. A profiling algorithm categorizes annotations and automatically proposes inferred user interests. Each user can present himself with a set of categories that are visually shown on his profile. Resources annotated by SN users, automatically categorised, are suggested as items that users can drag and promote in their public profile for each detected user interest, as shown in Fig. 6.1.

Many factors influence users' continued intention to use SNs such as social interactions, knowledge expansion and targeted recommendations [84]. The users' desire of social interactions has been demonstrated to increase the number of *likes* and comments in [66]. The assessment of users engagement with content gives the opportunity to improve targeted services and recommendation. User propulsion at showing knowledge for self-promotion is used, for example, by platforms such as LinkedIn (Q&A) and StackOverflow as a mean to increase the quality and number of crowdsourced annotations but, to the best of our knowledge, there is not a study in the literature which confirms that user profile interfaces affect and improve user activity in SNs. In 6.3.1 we have conducted a controlled experiment to show how the user's effort to shape his public identity can be exploited to increase user's activity and production of crowdsourced annotations.

Visual saliency We propose the use of visual saliency in SN systems and interfaces at two levels: *i)* at the automatic annotation level to reduce the computational cost of processing all the frames; *ii)* at the interface level to propose to the users possible frames of interest. The SN prototype also features a salient frames carousel above each video to ease the addition of



Soundgarden is an American rock band formed in Seattle, Washington in 1984 by

Clinton "Clint" Eastwood, Jr. (born May 31, 1930) is an American actor, film director, producer,

Yann Tierse 1970) is a r France. His is split betw

Figure 6.1: User profile interface: the user can publish resources of interest dragging suggestions from the below to the above carousel.

crowdsourced comments. Videos are preprocessed to eliminate letterboxing (i.e. black bars in videos). Then, visual saliency maps are extracted for all the video frames. Maps are defined by a visual attention model which uses a dynamic neural network on multiscale image features computed with the iLab Neuromorphic Toolkit [103]. Salient frames in the video carousel are selected by identifying the peaks of saliency using the crest detection algorithm proposed in [154]. Automatic annotation is performed on frames selected computing the average saliency of the video and choosing those above the average, to have a dense sampling of video content.

Crowdsourced annotations Users can comment videos at frame level and add semantic references to Wikipedia entities using an autosuggest widget, as shown in Fig. 6.2. Wikipedia entities are also extracted automatically using entities detection.

A carousel of the most salient frames is also shown above the video player as a video summary. This facilitates fast and accurate annotations at exact

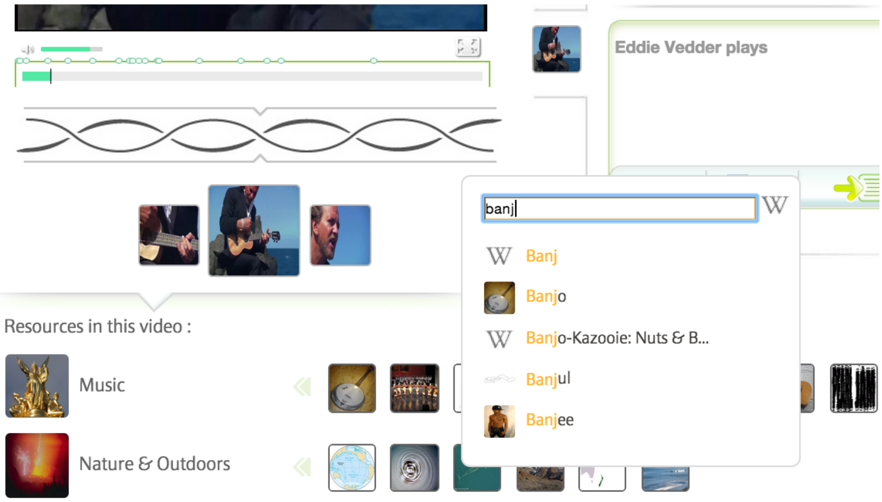


Figure 6.2: Wikipedia annotation in video frame-level comments.

timecodes, since users are more likely to interact with salient frames rather than with the less visually interesting ones. A vector of categories C, with the same dimensionality of the SN categories taxonomy, is used to represent video content. Each category in C is assigned with a weight defined by the average of the semantic distance of each annotation to the categories' taxonomy. This semantic relatedness between the terms is obtained using the Wikipedia Link-based Measure [97].

Visual features Automatic annotation of all the frames of the videos in a SN is a time-consuming task which requires a lot of resources. In the proposed SN video frames are subsampled according to their visual saliency, allowing the system to scale while maintaining a reasonably dense sampling of video content. The convolutional network used was trained on the ImageNet ILSVRC 2014 dataset to detect 1000 synsets. A very deep CNN with 16 layers [22] was used to extract the final output layer for each frame, containing 1,000 object probabilities. Video content is represented using a Bag-of-Words (BoW) approach. The features vector is computed using the frequency of occurrence of detected concepts with a probability above a threshold, then also complemented by crowdsourced annotations.

The Recommender Compared to user-based CF approaches, item-based recommenders minimise the sparse item ratings issue, are scalable and in general perform better than user-based recommenders [127]. The proposed hybrid RS adopts a solution that combines a semantic pre-filtering of content with an item-based algorithm. Videos are represented using a feature vector that concatenates the histogram of the categories of the crowdsourced comments and the BoW description obtained using the CNN classifier. User's rating on a video is computed combining explicit and implicit activity. Users can explicitly vote a video on a 5 point scale with a visual widget. Number of visualizations, frame browsing and annotations are also taken into account. In order to reduce the dimensionality of the item-item matrix used by the algorithm, a pre-filtering on the set of possible videos to suggest is performed. Given a user u, we extract a set F_u of videos for which u generated a rating. For each video v_i contained in F_u, the system selects the top-N similar videos creating a subset of similar videos S_i. The set of videos that will be used for the item-based recommender for user u is then composed by the union of all the subsets S_i, namely:

$$R_u = \bigcup_{i=1}^{|F_u|} S_i. \tag{6.1}$$

The set of video $R_u \cup F_u$ is used to create the item-item matrix used for recommendation. This set is significantly smaller than the whole collection of videos contained in the system. The pre-filtering step uses several approaches in order to infer the top-N similar videos. These approaches, reported in Sect. 6.3.3, exploit automatic and crowdsourced annotations as well as visual saliency, and use distance measures to compute the overlap between histograms distributions.

6.3 Experimental results

Recommendation is a prediction problem: the system should be able to predict the user's level of interest in specific items (e.g. videos) and rank these according to their predicted values [99]. In order to evaluate the accuracy of the prediction, a percentage of the collected data, represented by users ratings on videos, is extracted and used as test data, not used to train the RS. The RS produces rating predictions for the missing test data, that are compared to the actual values in order to evaluate the accuracy. The performance is evaluated using Root Mean Square Error (RMSE). The more accurately the RS predicts user ratings, the lower the RMSE will result. The SN dataset is composed by 632 videos, of which 468 have been annotated with 1956 comments and 1802 annotations. 613 videos were rated by 950 of the 1108 users of the prototype SN.

6.3.1 User profile interface

An A/B test experiment was conducted at the interface level to test the user profile influence on users' comments activity. The experiment was run on all the active users of the prototype SN for three months. Users were exposed to one of two variants of the SN, featuring (i.e. the variant) or not (i.e. the control) the profile curation interface. The variant was introduced in the third month, so that the number of users exposed to the variant is smaller than that of the control interface. Users who logged into the system and commented on videos since the third month were assigned to the variant (group B), whilst the others were assigned to the control group (group A). In this period of time there were 464 active users (321 in group A and 143 in group B) with a conversion rate of 3.75 and 5.81 average comments. User annotation average increased by a factor of 2.06. The result was statistically significant and validated by a *t*-test that gave a *t*-difference = -2.684. Minimum sample size for the evaluation criterion validity was calculated and resulted in 127 for both group A and group B with an optimum allocation ratio of 3.42. Results show a positive correlation between the use of the user profile interface and the increment in user annotations, and suggest that modules for profile curation can be effective in improving conversion rate in user online activity (e.g. videos annotations).

6.3.2 Visual saliency and manual annotations

The impact of the visual saliency of video frames on user comment activity has also been tested. In the experiment were considered: *i)* the number of comments added without using the most salient frames carousel and *ii)* all the comments, i.e. adding also those coming from a click in the carousel. Results of case *i* show that 53.5% of user comments are on frames with a saliency above the average saliency of the videos, and that the percentage of frames above the average saliency is 46.5%. Therefore, salient frames receive more attention by users, although not considerably. Results improve consistently considering also carousel driven annotations as in case *ii*: in fact the percentage of comments increases to 65.24%. Percentage of comments carousel driven is 24.01% of the overall dataset, showing that one out of four comments are added using the carousel: it is an high percentage considering threaded comments, added by users in response to others. So, it can be said that salient frames suggestion can be useful if proposed in a web interface as to visually capture the user's attention and help in the annotation tasks.

6.3.3 Recommendation

The RS is evaluated, in terms of RMSE, comparing it to several baselines: *i)* standard item-based RS, that considers users ratings of all the videos; *ii)* RS working on a selection of videos, based on similarity computed using system categories only (no BoW content description); *iii)* RS working on a selection of videos, based on content similarity (i.e. automatic annotations) computed on n randomly selected frames; *iv)* RS working on a selection of videos, based on content similarity computed on n frames with visual saliency above the average; *v)* RS working on a selection of videos, based on content similarity computed on *a)* frames with visual saliency score above the average and *b)* crowdsourced annotations.

Results are reported in Fig. 6.3 and show how the proposed *v)* approach results in a lower RMSE value than all the other approaches. In particular, it can be observed that video representation using salient frames improves over random selection, and that the addition of semantics extracted from manual annotations provides another improvement. In this experiment the threshold used to select the confidence scores of the classifiers is 0.85. In a second experiment we have evaluated the effect of the confidence of the classifier, using a threshold of 1. In this case the RMSE is further improved from 0.97 to 0.86.

6.4 Conclusions

In this Chapter we presented a system to improve an item-based video RS. The RS uses a reduced item-item matrix, computed from content based description of videos obtained from crowdsourced and automatic annotations. User engagement through profile curation and visual saliency has been used *i)* to increase the number of crowdsourced annotations, presenting the most relevant frames to users, and *ii)* to address system scalability in terms of

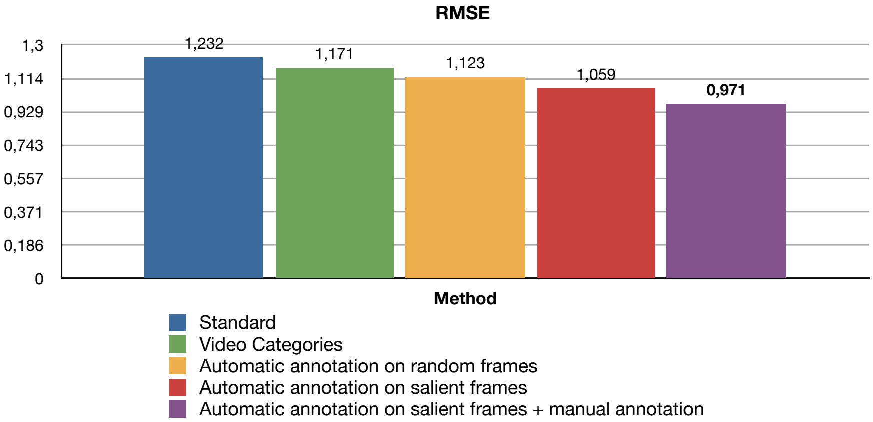


Figure 6.3: Comparison of the proposed recommender (rightmost result) w.r.t. baselines in terms of RMSE.

automatic annotation, reducing the number of frames to be processed. The effectiveness of exploiting human factors for user engagement (i.e. self-esteem in user profile interfaces and visual saliency) is evaluated by user experiments on a SN prototype. Experiments show also that the proposed RS improves over the standard implementation of an item-based algorithm, and that the combination of manual and automatic annotations is more effective than the use of a single type of annotations. A positive correlation of the two human factors with the performance of the RS is not yet fully demonstrated, but it can be hypothesized and it is worth of further investigation.

6.5 Acknowledgments

This work is partially supported by the “Social Museum and Smart Tourism” MIUR project (CTN01_00034_231545).

Chapter 7

Co-located Users Contextual Social Networking and Recommendation

In this Chapter we propose methods to be used in contextual social networks for the recommendation of potential friends, local experts and targeted services. The recommendation is based on an hybrid approach which combines content-based techniques, collaborative filtering and social media analysis. We also introduce the concept of co-located social network, explain why and how it can improve user engagement and, finally, we present the results in a mobile web-based social network application available only in the check-in area of an airport. The friend prediction algorithm has been evaluated through a user study on the demo application to measure the effectiveness of the recommendation. [1]

7.1 Motivations and Previous Work

With regard to context-based applications, the concept of context can be referred to where a user is, who she/he is with and what resources are nearby [130]. Contextual social networks can be seen as a variant of social

[1] Part of this Chapter has been published as "PITAGORA: Recommending Users and Local Experts in an Airport Social Network" in *Proceedings of the 23rd ACM international conference on Multimedia, pp. 755-756, 2015* [48].

networks where information about context is incorporated into the social network services [157]. Knowledge that can be extracted from online social network (e.g. Facebook and LinkedIn), along with information about the location, can be exploited in order to build a variety of contextual services, such as recommendation of people and activities.

In this Chapter we propose techniques of data analysis from social media to enhance friend and local experts recommendation for a social network only available in the subnet of a building, in the specific in the check-in area of an airport. The demo of the social network is a mobile web application we developed to collect data, test and validate the system. The proposed architecture, however, can be applied also in other domains and real locations where social networking applications could be used to offer contextual services. Contextual social networking techniques can be effective inside an airport, but also elsewhere, as better as the needs of visitors can be identified. In the check-in area of an airport people basically has to optimise their time and have the opportunity to get in touch with other people. Usually passengers are in airport for business trips or leisure and, in both cases, it has been demonstrated that they are more open to new experiences and social interactions than usual. In fact, some experiments [124] has been conducted that show how strangers in airport departure lounges have an higher degree of disposability to self disclosure.

Recommendations systems for friend prediction are very popular on the web but they still lack a footprint that can link social network users, temporarily co-located in the same place, to the services and opportunities of the place itself. Also the so-called contextual social networks are somehow ephemeral, based on general data such as social relation, interaction data, individual preference or interpersonal influence [71]. These networks foster meaningful interactions based on real relationships, shared interests, activities but can acquire a real value only taking into account the real intentions of users, especially in a situation where a lot of people is co-located .

In the case of an airport lounge, for example, the destination of a passenger, the type of trip (professional or not professional) or the shops present in the building are definitely essential infos from which friendship prediction or content recommendation based on user profiles can really benefit. A lot of works have been published in the last years addressing the problem of recommendations in Location-based Social Networks (LBSNs) using content based recommendation and collaborative filtering both for the suggestions of potential friends and the discovery of popular users (e.g. local experts, opinion leaders). At this regard content-based recommenders are costly, need maintenance, don't consider social opinion but are a valuable solution in particular to solve the cold start scenario. Collaborative filtering instead considers similar user preferences and uses a similarity model for score prediction. If it solves issues related to maintenance and availability of community opinion informations, suffers from possible issues due to data sparsity and system scalability [7]. Friends recommendation can be achieved analysing online user profiles and social graph interactions [24]. In this respect common connections or users degree of separation can be a valuable information especially if users accessing the network are co-located in the real world. Other Standard recommendation approaches are based again on content based analysis and collaborative filtering but also considering the social graphs and social interactions. A matrix factorization method which uses individual preference and interpersonal influence to improve the accuracy of social recommendation is proposed in [71]. The work stresses the fact that social influence is a powerful force which governs the dynamics of a social network. Some works exploit user location histories for recommendation. A correlation between different places and cities visited by users is computed on the basis of the trajectories followed by several people in local areas in [148] considering location-location distance and sequential ordering of visiting patterns. Zheng et al. [170] recommend local experts in a city analysing user location histories as GPS trajectories through a Hypertext Induced Topic Search inference model over a Tree-Based Hierarchical Graph. At this end user's geo-tagged social media content has also been exploited [3].

Recommendation approaches we propose are exploited and validated in a mobile contextual social network designed to be used by passengers inside of an airport structure: the network allows the user to obtain infos on his flight (e.g. status, time to gate close etc.), to interact with other passengers sharing his volatile or professional interests (e.g. to chat) and to receive personalised recommendations based on his/her preferences. The main goal is to improve the users travel experience bridging the gap between physical and virtual world exploiting traces and data available in Location Based Social Networks (LBSN) or collected through the application. Users can connect through Facebook and LinkedIn and their online data is used to build users' interests profiles and to recommend similar people and services. Destination information is exploited to suggest local destination experts present at the airport or on the same flight; retails semantics are used to suggest activities and places for dating through profile content analysis and matching between the users' mutual interests.

The remainder of this Chapter is organised as follows: Section 12.2 describes the overall system, focusing on the two main recommendation modules: friends prediction in Section 7.2.1 and local experts suggestion in Section 7.2.2. Evaluation and experiments on the recommendation system are proposed in Section 7.3.

7.2 Contextual social network

The end user interface has been developed as a mobile web application to be used in the context of a lounge of an airport. The goal of the application is to improve passengers experience providing a social network of co-located users able to recommend people and services contextual to the location and the user intentions. The application allows the user to search and display flight infos, to check the presence of other passengers on his same flight and to communicate with them through a real time chat. The main purpose of the application is to enhance the social interaction between users that are present in the airport at the same time. Recommendations are mainly constituted by suggestions of other users who share the same interests as well as of services within the airport (e.g. retails). The application also provides the passenger with a recommendation system of local experts, present at the airport, on the basis of his/her flight destination. The system uses an hybrid approach to the recommendation problem combining content-based filtering and collaborative filtering. Hybrid approaches have been proved to give better results than content-based filtering and collaborative filtering techniques solving issues related to the large amount and the sparsity of the data [44].

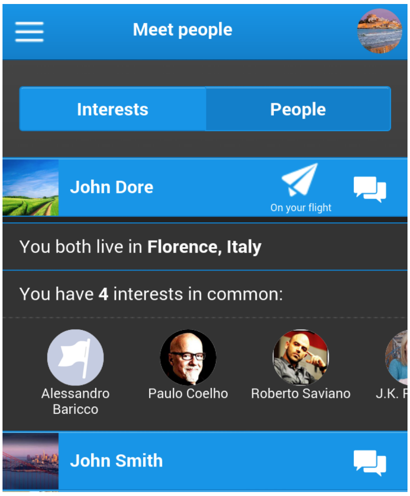
7.2.1 Users Recommendation and friends prediction

Users are profiled analysing data extracted by their Facebook and LinkedIn accounts and represented as a graph of users, interests and demographic infos modelled with hierarchical relationships. The different nature of this data is exploited in order to obtain recommendations concerning leisure (Facebook) and professional aspects (LinkedIn). When the user logs into the system using one of the two possible networks the following data are extracted:

- demographic infos, level of education, job history (Facebook and LinkedIn);
- photo albums (Facebook)
- groups and companies followed (LinkedIn)

.

Users' recommendation in the airport mobile contextual social network, is presented in two ways: as a list of possible friends and as a list of categories of interests extracted from Facebook that groups those friends (see Fig. 7.1).



You both live in **Florence, Italy**

You have **4** interests in common:

Figure 7.1: Recommendation of friends as proposed in the application, along with common items and interests.

Friends recommendation is based only on Facebook profiles but the same approach could be used also for any other social network. Profiles are described as vectors of pages on which users have expressed a 'like'. Although a standard collaborative filtering approach could have been used to estimate a neighbourhood of similar users, a main issue has to be addressed that is the sparsity of the dataset: in fact the number of Facebook pages is much bigger than the number of 'likes' that a single user can express. In order to solve the sparsity problem, the user recommendation module uses the co-occurence matrix approach to estimate additional possible items of interests. Users' recommendation is then achieved with a standard user-based algorithm considering the distribution of user interests and computing a users' neighbourhood with the Euclidean distance. Sparsity reduction is performed offline, using an item-based algorithm. Initially, a vector of user's preferences P is created. User's preference for each page is boolean: 0 if the user doesn't like the page, 1 otherwise. A matrix M of co-occurrence of 'likes' on pages is created for all the users in the system.

For example, given the following matrix M of co-occurrence calculated for 7 items and a vector of preferences P_u for a user u, the product rows to columns between M and P_u returns a vector R_u containing the inferred preferences.

$$\begin{array}{c} \\ i_1 \\ i_2 \\ i_3 \\ i_4 \\ i_5 \\ i_6 \\ i_7 \end{array}
\begin{array}{c} \begin{array}{ccccccc} i_1 & i_2 & i_3 & i_4 & i_5 & i_6 & i_7 \end{array} \\
\begin{pmatrix} 5 & 3 & 4 & 4 & 2 & 2 & 1 \\ 3 & 3 & 3 & 2 & 1 & 1 & 0 \\ 4 & 3 & 4 & 3 & 1 & 2 & 0 \\ 4 & 2 & 3 & 4 & 2 & 2 & 1 \\ 2 & 1 & 1 & 2 & 2 & 1 & 1 \\ 2 & 1 & 2 & 2 & 1 & 2 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix} \end{array}
* \begin{array}{c} P_u \\ \begin{pmatrix} 1.0 \\ 0.0 \\ 0.0 \\ 1.0 \\ 1.0 \\ 0.0 \\ 1.0 \end{pmatrix} \end{array}
= \begin{array}{c} R_u \\ \begin{pmatrix} 12.0 \\ 6.0 \\ 8.0 \\ 11.0 \\ 7.0 \\ 5.0 \\ 4.0 \end{pmatrix} \end{array}$$

In this example, rating for items i_2, i_3 and i_6 are inferred using an item-based similarity, even if the user never explicitly added a preference. The final vector representing user u is created starting from R_u using an higher weight for the items that explicitly were preferred by the user (i.e. items that have value 1 in P_u) and a lower weight for the inferred items. To create an ordered list of suggestion for the user u we use Euclidean distance on the preferences vector, finding the nearest-N users. An additional score is finally added for users that share the same demographic data or professional history. Performance of the users recommender is evaluated in Section 7.3.

7.2.2 Local Experts Recommendation

The identification of local experts is obtained through the analysis of social media extracted from Facebook. In particular user travels and location history are computed analysing geo-tagged pictures and identifying photo albums, birthplaces and places of residence and exploited to produce destination-based recommendations.

The mobile app of the contextual social network provides a way for users to select his/her flight and destination. The goal of the local experts recommender is to suggest users that have an high level of travel experience about the flight city destination or region.

The proposed method for local expert computation is based on the approach proposed in [170], where human location history is exploited in order to recommend cities Point-Of-Interests. Our approach complements this computation considering not only the user travel experience but also the correlation and the distance between cities. Given a set of users U and a set of visited cities C, a matrix V is defined where the item v_{ij} of V represents how many times the user u_i has visited the city c_j, with $0 \leq i < |U|$ and $0 \leq j < |C|$. Visited cities are extracted from user activity in the social network application: we consider the selection for a flight destination on the mobile application as a visit.

The *travel experience* vector E of users is calculated iteratively on the basis of the number of travels in cities as it follows:

$$E_n = E_{n-1} \cdot V \cdot V^T \tag{7.1}$$

In (7.1) E_n indicates the vector E at the n iteration that is initialised with $E_0 = (1, 1, \cdots, 1)$. The vector E is then normalised, dividing by its highest value.
The correlation $CORR(c_i, c_j)$ between two cities c_i and c_j is expressed as:

$$CORR(c_i, c_j) = \sum_{u_k \in U'} \alpha * e_k \tag{7.2}$$

where U' represents the group of users who has visited both c_i and c_j, e_k is the component of E relative to the user u_k. The weight factor α, with $0 < \alpha \leq 1$ is defined taking into account the Euclidean distance between latitude and longitude of the two cities.

Finally, we need to compute user's level of experience of a user u for a given city c. To this end, we define n_k as the number of visits of the user u in the city c_k, based on previous activity on the network (i.e. previous flight destinations). We also extract from Facebook the number of geo-tagged photos p_k that the user has published in c_k .

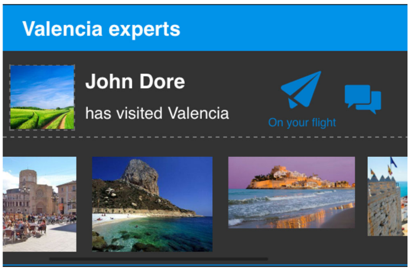


Figure 7.2: Local Experts Recommendation. Photos published in Facebook albums and used in the recommendation are shown. The expert will be on the same flight and users can chat.

From these data, a rating $exp_{u,c}$ can be assigned to user for a city as:

$$exp_{u,c} = \sum_{k=0}^{|C|} (1 + p_k) * n_k * CORR(c, c_k) \tag{7.3}$$

We also consider if a user lives (or has lived) in c or if she had educational/professional history in the city, adding points to the rating $exp_{u,c}$. Local experts whose score is greater than a threshold are shown as a list of recommendation, in descending order, in a dedicated view of the mobile app as shown in Fig. 7.2.

7.3 Evaluation

The proposed friends recommendation system is based on the ranking of user similarities and can be therefore seen as an information retrieval system, considering a user as a query term. To evaluate the relevance of recommended potential friends we have exploited the normalized Cumulative Discounted Gain (nDCG) [69] measure. The intention of the evaluation is to compare the generated list of recommended people with the ideal list created from a relevance score given by the user. We collected a ground truth asking 150 users to express a relevance score (on a 0 to 3 scale) for the first J people suggested by the system when they use the application. For example, for a suggested list of people ordered by the recommender as (P1, P2, P3, P4), the user provides as ground truth the relevance vector of scores (1, 3, 2, 0). For each list of recommended potential friends for these users, we can obtain a score list where the scores are provided by ground truth. Assuming each user u expressed a relevance r_{u_j} from being recommended an item j, the average Discounted Cumulative Gain (DCG) for a list of J items and N users is computed as

$$DGC = \frac{1}{N}\sum_{u=1}^{N}\sum_{j=1}^{J}\frac{r_{u_j}}{max(1, \log_2 j)} \tag{7.4}$$

In this experiment, a logarithm with base 2 is used to ensure all positions are discounted. The nDCG is the normalized version of DCG given by

$$nDCG = \frac{DCG}{DCG^i} \tag{7.5}$$

where DCG^i is the ideal DCG, computed on the model distribution of relevance depending on the number of items J as shown in Table 7.1. We calculated nDCG for the top-J item, resulting in values of 0.767 with j=5 and 0.872 for j=10.

Table 7.1: Ideal relevance and computed nDCG for top-J recommended items

J	**Ideal Relevance**	**nDCG**
5	(3,3,2,1,0)	0.872
10	(3,3,2,2,2,1,1,0,0)	0.767

7.4 Conclusions

In this Chapter we have described mechanisms of friends prediction and recommendation, and suggestion of local experts for a contextual social network designed to be used in real-time by passengers in the check-in area of an airport. We motivate the work with the need, for a social network that pretend to be contextual, to adapt itself to services and intentions of volatile and co-located groups of persons, such as a lounge of an airport. We propose a novel approach to the refinement of friends recommendation strategies which takes into account profiling techniques from social network analysis (e.g. Facebook and LinkedIn) and inferred ratings for users' neighbourhood detection and identification. A method for local experts recommendation is also proposed which improves from previous work exploiting social media analysis. Use cases are shown through a developed mobile web app by which users can search for flights, destinations and facilities infos and receive targeted services and recommendations. Finally we evaluate the recommendation and the effectiveness of the overall system with a user study based on the nDCG measure which shows good results.

Chapter 8

smArt: Open and Interactive Indoor Cultural Data

In this Chapter we present smArt[1]*, a low-cost framework to quickly set up indoor exhibits featuring a smart navigation system for museums. The framework is web-based and allows the design on a digital map of a sensorized museum environment and the dynamic and assisted definition of the multimedia materials and sensors associated to the artworks. The knowledge-base uses semantic technologies and it is exploited by museum visitors to get directions and to have multimedia insights in a natural way. Indoor localisation and routing is provided taking advantage of active and passive sensors advertisements and user interactions. In this way we overcome the Global Positioning System (GPS) unavailability issue in indoor environments.* [2]

8.1 Context and Motivation

The problem of linking physical spaces with structured data is urgent considering from the one hand the opportunities offered by the evolution of the semantic web and from the other the increasing adoption of the so-called

[1] Demo video: https://vimeo.com/miccunifi/smart

[2] This chapter has been published as "smArt: Open and Interactive Indoor Cultural Data" in *Proceedings of the 23rd ACM International Conference on Multimedia, pp. 807-808, 2015* [55].

Internet-Of-Things (IOT). Furthermore, managers in the cultural heritage need easy to use tools to promote, curate and publish cultural data that may be exploited at several levels by students, tourists, professionals, researchers and so on. To this end tools have to provide museum curators with facilities for artworks' search, browsing and collection, and not at least the opportunity to make cultural resources available as public structured data on the web. At the same time, nowadays, real spaces can easily be made reactive and low-cost solutions are desirable. smArt fulfills these requirements and provides a tool where cultural resources can be browsed and published, from and on the web, using a big and extendable repository (i.e. DBPedia dataset[3]). Data can be enriched with sensors information for *ad hoc* deployable installations. In details, the web application allows to associate artworks in a semi-automatic way with different types of sensors and features proximity sensing and routing in an indoor environment. This is not trivial: in fact while tourism electronic guides for outdoor are widespread and provide access to contextual multimedia data relying on GPS technology, things are more difficult in indoor where GPS is not available. Sub-room indoor localisation is an active area of research which includes applications in reactive indoor spaces. smArt exploits Bluetooth Low Energy (BLE) beaconing, in synergy with other tools, as an indoor positioning and routing technology. Bluetooth Estimote Beacons[4] and automatic generated QR codes can be automatically and manually associated to artworks or locations in the knowledge-base and then physically positioned. Cited technologies are low or without cost and don't involve an infrastructural overhead: two common requirements for cultural public institutions.

8.2 The System

The system is mainly composed by two modules: 1) a web-based application for the semi-automatic ingestion and management of Open Linked Data regarding the cities of Venice, Rome and Florence in Italy; 2) a mobile Android application which exploits the data generated by the web app and reacts to the expected signals in the real environment.

[3] See http://bit.ly/1HxOUVI
[4] See http://bit.ly/1d7dZdB

8.2.1 Configuring the environment: the Web Application

The web application is used for the ingestion, creation and management of data concerning museums. It provides a graphical user interface for building data and components mashups in order to configure a sensorized environment. Users are enabled to 'pipe' several interface components and then set up rules for how content should be modified. There are three main components: 1) City Component: it allows to choose cities from which to select public multimedia data; 2) Museum Component: it is used to collect, aggregate and enrich data about museums and artworks from DBPedia through Open Linked Data; 3) Sensor Component: it can be applied to artwork collections in order to automatically associate physical sensors. Components are managed and organised on the interface using a drag-and-drop paradigm. A component is shown as a circle icon with a label and an image. When dragged on other components icons can make appear contextual menus in order to apply modifiers. The Museum Component provides contextual panels to search museums, select and associate sensors to artworks (i.e. Beacons or QR codes). Furthermore, for each museum the user can: 1) interactively draw the museum map of the environment to be sensorized; 2) decide the location of the artworks; 3) define the access point to the museum halls and finally 4) mark out all the trajectories that visitors can use to reach the artworks. All this infos are then used by the mobile application to provide localisation and routing systems to museum visitors. The web app has been developed in HTML5 for the client and uses PHP and MySQL on the server for metadata storing. Storing and communication with the semantic knowledge-base is performed through RDF and Sparql Queries to a self-hosted DBPedia endpoint. The knowledge-base is reachable on a self-hosted Virtuoso Server[5] and it uses triple-store dumps of Wikipedia frequently and automatically updated.

8.2.2 Mobile App, Localisation and Routing

smArt mobile application allows the visitor to localise herself in the indoor museum and provides a routing system to guide the user to artworks of interest on the basis of the data generated by the web application. The app has been designed with the aim to enhance the user experience of a visitor

[5] See http://bit.ly/1BqKEVP

approaching or searching for an artwork and provides multimedia insights exploiting natural interaction paradigms. It is well known that GPS is not working in indoor locations due to the poor signal coverage. Furthermore, indoor localisation is particularly challenging for several reasons: presence of obstacles and moving people, interference caused by other electronic devices etc. Standard solutions contemplate active (QR code scanning, NFC) and passive sensors (beacon bluetooth for proximity detection or triangulation for exact location). smArt exploits bluetooth beacons which are cheap and well supported and require a low level of interaction. As an alternative each artwork can be automatically or manually associated to QR codes which have no cost but need more user participation.

The app has been developed as an Android application and uses an SQL Lite database generated by the web application and stored on a server. The interface has been designed following the Google guidelines for material design. The main interface is map based and provides outdoor navigation. Through a sliding-up panel the user can check his localisation and browse all the nearby museums where an interactive exhibit has been set up. Google directions are also provided. When a user approaches a museum she is notified on the app interface of the possibility to switch to an indoor map visualisation. The map is rendered in realtime on the device using canvas and vector shapes. Zoom and drag gestures are enabled. Artworks equipped with a sensor are visualised on the map as icons. Once the user has localised herself or the app has identified her location, the user can select any artwork on the map or using the sliding panel in order to be suggested with the shortest path to it. User location is acquired when the app receives the unique identifying information broadcasted by a beacon via bluetooth or when the user scans a QR code associated with an artwork or provided as a localisation hotspot in the museum. A background service is always active and listens to BLE advertisements. When the device receives the signal, the user is notified and contextual artwork multimedia data are shown.

The indoor engine is in charge to draw and manage the map and the navigation system: paths are computed modelling the information about museum rooms and artworks as a graph of traversable spots. Spots have been defined and arranged by the web app user on the map and can be sensorized artworks, path spot, door spot or museum visitor localisation spots. The shortest path to an artwork is estimated on the fly by the indoor engine using the Dijkstra's algorithm and then visualised on the map. The engine provides also a completely automatic system to calculate the shortest path in the case that the web app user has not marked the path spots required to navigate from an artwork or a localisation in a room to another. This is achieved using automatic 2D polygon convex partitioning of the museum map: the Hertel-Mehlhorn algorithm is exploited which is never worse than $2r + 1$ pieces, where r is the number of reflex vertices. Once the polygon is partitioned the center of mass of each partition is identified and treated as a path spot to be used in the graph by the Dijkstra's algorithm to build the itinerary from the user position to the artwork of interest. The algorithm allows to face situations where the map is a regular polygon but paths from an artwork or a localisation in a room to a target artwork in another room could cross the museum hall walls resulting in a wrong feedback to the user.

8.3 Conclusions and Future Work

In this Chapter we present a web-based framework designed for museum curators to manage and set up easily interactive exhibits in a sensorized environment. Museums, artworks and associated multimedia materials are retrieved from and saved to the web using Open Linked Data. Artworks can be placed on the map and associated with cheap and easy to install actuators to be placed in the real museum environment. Exploiting these data museum visitors equipped with an *ad hoc* mobile app can enjoy interactive exhibits in an effective and natural way. Future work will focus on the refinement of methods for the reduction of localisation errors using BLE technology. A good starting point are the results in [91].

Chapter 9

Imaging Novecento. A Mobile App for Automatic Recognition of Artworks and Transfer of Artistic Styles

Imaging Novecento is a native mobile application that can be used to get insights on artworks in the "Museo Novecento" in Florence, IT. The App provides smart paradigms of interaction to ease the learning of the Italian art history of the 20^{th} century. *Imaging Novecento* exploits automatic approaches and gamification techniques with recreational and educational purposes. Its main goal is to reduce the cognitive effort of users *versus* the complexity and the numerosity of artworks present in the museum. To achieve this the App provides automatic artwork recognition. It also uses gaming, in terms of a playful user interface which features state-of-the-art algorithms for artistic style transfer. Automated processes are exploited as a mean to attract visitors, approaching them to even lesser known aspects of the history of art. [1]

[1] This Chapter has been published as "Imaging Novecento. A Mobile App for Automatic Recognition of Artworks and Transfer of Artistic Styles" in *Digital Heritage. Progress in Cultural Heritage: Documentation, Preservation, and Protection: 6th International Conference, EuroMed 2016, Nicosia, Cyprus, October 31 – November 5, 2016,*

9.1 Introduction

Modern museums can provide new paradigms for experiencing artworks. Thanks to the technological development, novel initiatives include pervasive uses of tech to create interactive experiences for visitors throughout a museum. However, making content relevant and appealing through these modern technologies is a difficult problem, requiring more and more interactivity as the audience is shifting towards a 'multimedia point of view'. Moreover, while the massive amount of available artworks constitutes a huge resource for education and recreation purposes, it can also be a cognitive burden for visitors.

The cognitive process related to learning has been an active subject of study in recent decades. According to cognitive load theory, learners must cope with a certain level of cognitive effort to process new information [112]. In this regard, multimedia education, defined as "presenting words and pictures that are intended to foster learning" [94], can be an effective remedy because it facilitates the activation of sensory and cognitive perceptions (e.g. visual and notional memory), avoiding visitors from information overloading. This can also be reinforced by gamification, that is the use of playful experience to help a user find personal motivations and engagement with serious content [122]. This combination can enhance the visitor's involvement and further lower its cognitive effort. Using gamified applications, museum visitors have the opportunity to feel the emotion of a game, share results with friends on social networks or become part of a game community [105]. This aspect of learning through gaming is even more valuable in the context of the "Bring Your Own Device" (BYOD) approach [5] that allows on demand access to digital content on personal devices. The BYOD approach and gamification have been identified in the NMC Horizon Report 2015 to be increasingly adopted by museums in one year's time or less for mobile and online engagement [73]. In this Chapter we report our experience in embedding these concepts into *Imaging Novecento*, a system built around a mobile application developed for the museum "Museo Novecento" in Florence, IT. We aimed at improving the learning process of the visitors by exploiting a simple gamification paradigm, and at reducing visitors cognitive load. To this end, we also developed a state-of-the-art computer vision system that is able to 1) recognize artworks from photos; 2) apply their style to user

Proceedings, Part I, pp. 781-791, 2016 [9].

photos.

9.1.1 The Museo Novecento in Florence and Innovecento

The “Museo Novecento” in Florence, IT, is a museum opened on June 24, 2014. The museum is dedicated to the Italian art of the 20^{th} century and offers a selection of about 300 artworks distributed in fifteen exhibition halls on two levels. The venue is located in the former hospital of the “Leopoldine” in Piazza Santa Maria Novella. The museum has been an example of innovation since its genesis, thanks to the prompt adoption of the latest multimedia technologies.

In March 2015, in order to improve the visitor experience, the Municipality of Florence has published an open call “INNOVecento - Novecento Museum Innovation Lab” inviting companies and professionals to propose ideas and solutions based on ICT.

Five companies specialized in technologies applied to cultural heritage have already responded to the call which, at the time of writing, is still open.

As NEMECH, centre of competence of the Tuscany region in Italy, we proposed *Imaging Novecento.* The App features automatic recognition of artworks through the visitor’s smartphone and automatic transfer of artistic styles from artworks. These styles can be applied to user images.

9.1.2 Motivations and design

The target of the App is rather wide. Although *Imaging Novecento* can be used by anyone (e.g. tourists and residents), during the design process we identified a specific audience. We mainly target the App towards people in a relatively young age (between 14 and 30 years old), more accustomed to digital technologies, open to technological innovation and to gamification.

One of the main ideas of the App is to exploit the pervasiveness of mobile cameras in modern smartphones to reduce the cognitive effort required to museum visitors. In fact, despite themed rooms and the ubiquitous explanatory cards, users can still be overwhelmed by the great number of artworks present in the museum. Labels in museums can be very concise or, on the contrary, can be filled with lots of explanation, often generic, not highlighting salient features of individual paintings.

By using *Imaging Novecento*, the visitor can take a picture of the artwork he is interested in. The App will automatically recognize the painting and provide related information. Another reason for the adoption of this automatic process is the resistance of museums' curators to place or attach additional materials, such as QR codes or BLE iBeacon [64], next to artworks.

Furthermore, tourists and school groups are usually 'hit-and-run' visitors who tend to rapidly forget or do not have the time to process the overload of information. To solve this issue, *Imaging Novecento* leverages a playful feature that employs state-of-the-art algorithms for transferring artistic styles from recognized artworks to user images. This is done using a gamification paradigm at the interface level. Gamification techniques have been proved to be useful in engaging students in the learning process, improving their skills and maximizing their long-term memory [33].

9.1.3 Previous work

Several previous works have addressed the problem of providing an engaging experience to museum visitors. Rapid technological development has led to the implementation of a lot of applications. There are several active trends for virtual museums: immersive reality [56, 90], natural interaction installations [8, 42], mixed reality, mobile applications [23, 166]. While they all offer increasing engagement of visitors, only recently studies on the effects of audience have been carried out [73, 110]. In particular, a recent audience study has been conducted on the case of the "Keys to Rome" international exhibition, hosted at the "Imperial Fora Museum" in Rome in 2015, to assess the impact of these technologies on cultural heritage. The exhibition was made up of 11 digital installations and applications, installed in the museum [110]. The study highlights some fundamental aspects that must be taken into account when designing applications for virtual museums: 1) the majority of museum visitors are tourists and school groups; 2) visitors generally require applications with an high level of interactivity, particularly on their mobile devices; 3) it is essential for the UX design to use metaphors of informal learning capable to stimulate attention, memory and engagement (e.g. through gamification) in visitors.

Automatic artwork recognition Automatic artwork recognition is a long standing problem in applications for cultural heritage. Descriptors such as SIFT and SURF have been used for years in order to address this task [125, 140] due to their accuracy in recognizing paintings. Crowley and Zisserman [31] retrieve artworks finding object correspondences between photos and paintings by using a deformable part based method. More recent approaches for artwork recognition adopt Convolutional Neural Networks (CNN) as in [2], where a holistic and a part based representation are combined. Peng and Chen [113] exploit CNNs to extract cross-layer features for artist and artistic style classification tasks. Artistic style recognition is also performed in [75] on two novel large scale datasets. Similarly to these works, we explore the use of CNNs features but we aim to obtain a global representation that is semantically meaningful and also capable of retaining low level visual content information.

Artwork recognition has also been used with wearable devices, as in [8] where the user's position is jointly estimated with what he is looking at.

Artistic style transfer Regarding the application of artistic style to photos, a lot of research has been done in the past. The problem of rendering a given photo in the style of a particular artwork is known in literature as a branch of non photorealistic rendering [81]. This class of works use texture transfer [35, 162] to achieve style transfer. These techniques are non-parametric and directly alter image pixels of the content image into predefined styles. Another direction of work focuses on the idea of separating style and content in order to 'remix' them together in different configurations. First works were evaluated on much simpler images such as characters in different handwritings [141] or images representing human body configurations [36]. Only recently, the breakthrough paper from Gatys *et al.* [57] showed the possibility of disentangling the content from the style of natural images by using a convolutional neural network based representation.

The advantage of this approach is the capability of performing style transfer from any painting to any kind of content images. The approach was recently extended with a more advanced perceptual loss [72] and also applied to movies [1] by considering the optical flow.

9.2 The System

The system is composed by two main components: a mobile App and a computer vision system responsible to address the two tasks of automatically recognize artworks and apply artwork styles to user photos. The mobile App is used by the visitor in the museum and is the *fulcrum* of the user interaction. Once installed by the user in his mobile phone, it allows to take pictures, deliver artwork information and request the style transfer to new photos. Due to the limited amount of computational power available on most mobile devices, the computer vision system is deployed on a scalable web server system that processes requests from the mobile App. Since the two tasks use quite different technologies, we discuss them separately in the following sections.

9.2.1 The Mobile App

Imaging Novecento has been developed as an Android application using Ionic [2]. Ionic is a framework, based on Sass and AngularJS, for building highly interactive native web apps through mobile-optimized HTML, CSS and JS components and tools. *Imaging Novecento* is a contextual App that can be used exclusively inside the Museo Novecento in Florence. An information flyer of the App is delivered to the visitor at the ticket office. In the flyer there are a QR code, through which the visitor can download the App from the Google Play store, and the list of the artworks on which the App can perform the automatic recognition and style transfer processes. The list comprises a selection of twenty artworks for which the museum's curators have provided multimedia materials. The App interface (Fig. 9.1) is quite simple and is organized in two main views: 1) the Camera View and 2) the Artwork Details view.

The Camera View allows the visitor to frame one of the artworks on the list in order to have it immediately recognized by the automatic system. Proper feedback is given in case the recognition is not successful. Once the artwork is recognized, the Artwork Details view is activated. In this view, exhaustive but concise information about the author, the history of the artwork and its artistic style are given. An infographic is presented to the user. It works as a "call to action" for enabling the transfer of the recognized painting style to a photo from the user's device gallery. The infographic provides an animated preview that shows the result of the artistic style transfer on a predefined picture. After the image has been successfully uploaded, the remote process for style transfer is performed. The result of the elaboration is then sent to the user in a few minutes by email. The image

[2] `http://ionicframework.com/`

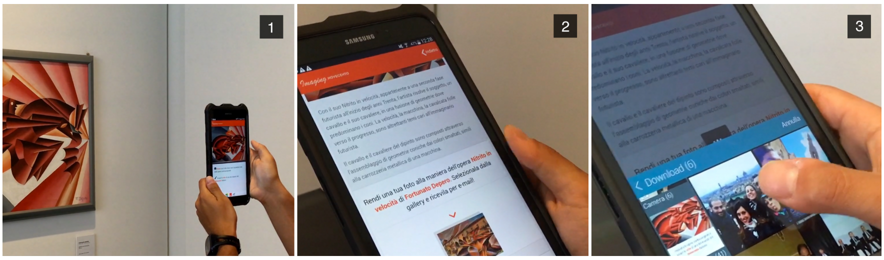


Figure 9.1: *Imaging Novecento* in action: 1) the user takes a picture of an artwork; 2) the artwork is recognized and insights are shown; 3) the user selects a photo from his own gallery in order to apply that artwork style and to share the results on social networks.

has a resolution of 900px wide preserving the original image aspect ratio and can be shared on the most popular social networks (e.g. Facebook).

9.2.2 Automatic Artwork Recognition

Artwork recognition is performed through a Python web server with a REST interface. The server processes the image and returns the ID of the recognized painting. The recognition step combines modern deep features with classical Support Vector Machines (SVM) in order to classify photos of paintings. Image features are extracted using a deep convolutional neural network (CNN), and are then evaluated using a set of classifiers, one for each recognizable artwork. The neural network we adopted is the Caffe reference model [70], fine-tuned for style recognition using the FlickrStyle dataset [75][3].

In order to obtain a representation which is at the same time semantically meaningful and capable of retaining low level visual content information, we extract image features from an intermediate level of the network. In particular, we adopt the *pool5* feature map, the latest one before the fully connected (FC) layers of the CNN. In fact, FC layers trade spatial information for a more semantic representation, which is highly coupled with the task and with the visual domain on which the network has been trained. This choice is therefore motivated by the fact that our visual domain, while being quite close, is different from the one of FlickrStyle. Moreover, since a sufficiently

[3] the network is available online at `http://caffe.berkeleyvision.org/gathered/examples/finetune_flickr_style.html`

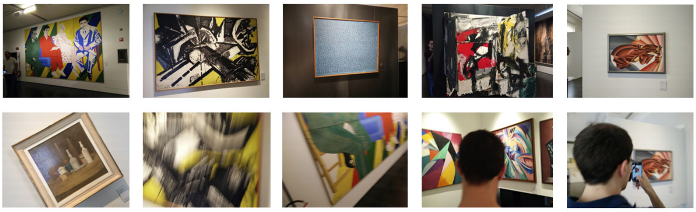
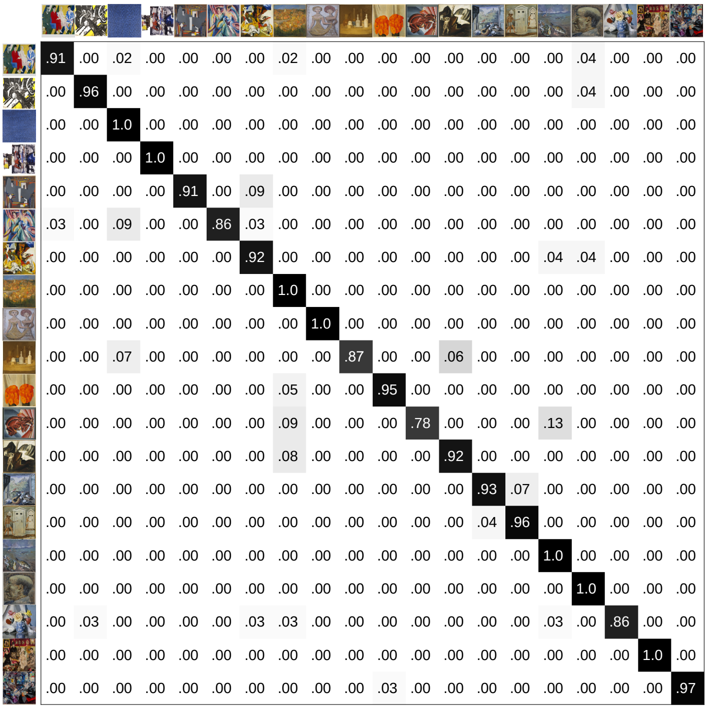


Figure 9.2: Samples from the dataset collected at the museum. On the first row standard pictures are shown, depicting the painting in their entirety. On the second row instead, are reported more challenging photos, due to blur, occlusion or rotation.



.91	.00	.02	.00	.00	.00	.00	.02	.00	.00	.00	.00	.00	.00	.00	.00	.04	.00	.00	.00
.00	.96	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.04	.00	.00	.00
.00	.00	1.0	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
.00	.00	.00	1.0	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
.00	.00	.00	.00	.91	.00	.09	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
.03	.00	.09	.00	.00	.86	.03	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
.00	.00	.00	.00	.00	.00	.92	.00	.00	.00	.00	.00	.00	.00	.00	.04	.04	.00	.00	.00
.00	.00	.00	.00	.00	.00	.00	1.0	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
.00	.00	.00	.00	.00	.00	.00	.00	1.0	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
.00	.00	.07	.00	.00	.00	.00	.00	.00	.87	.00	.00	.06	.00	.00	.00	.00	.00	.00	.00
.00	.00	.00	.00	.00	.00	.00	.05	.00	.00	.95	.00	.00	.00	.00	.00	.00	.00	.00	.00
.00	.00	.00	.00	.00	.00	.00	.09	.00	.00	.00	.78	.00	.00	.00	.13	.00	.00	.00	.00
.00	.00	.00	.00	.00	.00	.00	.08	.00	.00	.00	.00	.92	.00	.00	.00	.00	.00	.00	.00
.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.93	.07	.00	.00	.00	.00	.00
.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.04	.96	.00	.00	.00	.00	.00
.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	1.0	.00	.00	.00	.00
.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	1.0	.00	.00	.00
.00	.03	.00	.00	.00	.00	.03	.03	.00	.00	.00	.00	.00	.00	.00	.03	.00	.86	.00	.00
.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	1.0	.00
.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.03	.00	.00	.00	.00	.00	.00	.00	.00	.97

Figure 9.3: Confusion matrix for the artwork recognition module. Each row indicates the percentages of correct and incorrect classifications for a given artwork.

large dataset was not available to perform a further fine-tuning step, SVM classifiers have been trained to adapt the framework to the App's domain and be able to classify artworks correctly. For training the classifiers we used approximately 1,800 images, gathered at the museum using different smartphones and tablets, namely Galaxy S4, Galaxy Tab, iPhone 6, iPad Mini and OnePlus One. These images represent all of the twenty artworks plus a 'negative' set of images containing other scenes and paintings inside of the museum. They are used to reduce the false positive rate when the user accidentally attempts to recognize other paintings. All the classifiers are One vs All SVMs. During the evaluation phase, the ID of the highest scoring one is returned to the mobile App, if it scores above a cross validated threshold.

Details about the recognized artwork are then provided to the user, who can upload a personal photo to get the style of the painting transferred on to it. Calls to the web server are handled asynchronously and each request takes approximately 300ms on a CPU.

In order to test the recognition accuracy we collected an additional set of photos which were not used for training. For each one of the twenty artworks in our system, we collected approximately 30 photos taken from different viewpoints, with different scales and degrees of occlusion. Fig. 9.2 shows some of the photos from the test set. Some of them are "difficult" in a sense that might be blurred or taken from challenging viewpoints and artworks may be partially occluded by other visitors. Despite these difficulties our system achieves an overall good performance with a mean accuracy of 94.01%. In detail, in Fig. 9.3 we report the confusion matrix for the twenty artworks in the test set, showing how often each painting is correctly classified or confused with other artworks. As can be seen, the majority of artworks are perfectly recognized. Only four artworks have performance slightly inferior to 0.9, due to the difficult lightening conditions present in their specific locations at the museum.

9.2.3 Artistic style transfer

From the Artwork Detail view of the mobile App, the user has the possibility to upload a personal image on which the style of the artwork will be applied. In this way, entertaining personal pictures that share similarities with the artworks can be obtained and shared on social networks. As a result, a visit at the museum can become a playful experience, combining gaming and learning aspects for young visitors.

We base our approach on that of Gatys *et al.* [57], that is capable of freely mixing style and content of two different photos. The main advantage of this approach is its broad applicability to different styles, in contrast to fixed handcrafted styles [35, 162]. This allows a museum curator to easily add new artworks in the system without requiring the development of a new transfer style algorithm.

Following [57], our approach uses a CNN to derive a neural representation of content and style. The feature responses of a pre-trained network on object recognition (VGG-19 [133]) are used to capture the appearance of an artwork image and the content of a user photo under the form of texture information. We start from a blank novel image that is altered with back-propagation until its neural representation is similar in terms of euclidean distance to the style and content representations.

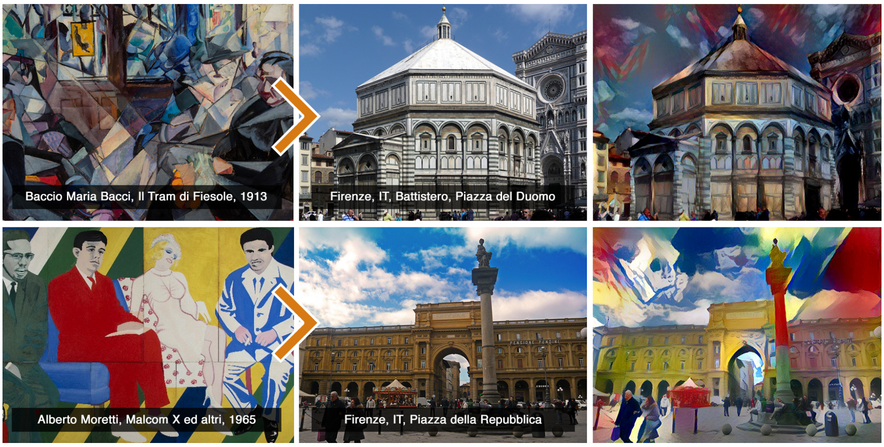

Baccio Maria Bacci, Il Tram di Fiesole, 1913
Firenze, IT, Battistero, Piazza del Duomo
Alberto Moretti, Malcom X ed altri, 1965
Firenze, IT, Piazza della Repubblica

Figure 9.4: Two examples of image stylization: 1) Baccio Maria Bacci, “Il tram di Fiesole”, applied to a picture of the Battistero in Florence, IT; 2) Alberto Moretti, “Malcom X ed altri”, applied to a picture of Piazza della Repubblica, also in Florence.

Unfortunately, the generation of the image is quite computational intensive. For an image of 900 pixel large, it takes about ∼90 seconds on a K80 NVIDIA GPU. As a result, the requests have to be handled offline since it is not possibile to obtain the output image in few seconds. Considering also that multiple requests can be made at the same time from multiple users, we implemented a scalable web server that is able to be easily deployed on several interconnected nodes. Web requests are handled in Python and enqueued to a distributed queue run by a Celery[4] server. By treating each request as a single unit task, it allows to process the images in a distributed batch fashion on several GPUs and several servers if available. After completing the computation, each output image is sent to the user via email, together with a description of the artwork. We also include links to share the image to several social media, with the aim of enabling viral publicity of the museum.

9.3 Conclusion

We presented the *Imaging Novecento* App, recently developed for the "Museo Novecento" in Florence, IT. Following previous studies on cultural heritage audience and applications, the App aims at enhancing the experience in the museum reducing cognitive load and exploiting gamification. The App automatically recognizes a selection of paintings and provides insights on artworks and their authors. The user can upload a personal picture with his smartphone to get it stylized with the recognized artwork style. He also has the possibility of sharing it on social networks. In the Chapter we show how computer vision technologies can be exploited to increase interactivity and reduce cognitive load. This can attract the targeted audience to the museum and further engage people with content.

Acknowledgments.

We acknowledge the support of the "Museo Novecento" in Florence, IT, and the Municipality of Florence, IT. This research was supported by "THE SOCIAL MUSEUM AND SMART TOURISM", MIUR project no. CTN01_00034_23154_SMST.

[4] `http://www.celeryproject.org`

Chapter 10

Deep Artwork Detection and Retrieval for Context Aware Smart Audio Guides

In this Chapter we address the problem of creating a smart audio guide that adapts to the actions and interests of museum visitors. As an autonomous agent, our guide perceives the context and is able to interact with users in an appropriate fashion. To do so, it understands what the visitor is looking at, if the visitor is moving inside the museum hall or if he is talking with a friend. The guide performs automatic recognition of artworks, and it provides configurable interface features to improve the user experience and the fruition of multimedia materials through semi-automatic interaction. Our smart audio guide is backed by a computer vision system capable to work in real-time on a mobile device, coupled with audio and motion sensors. We propose the use of a compact Convolutional Neural Network (CNN) that performs object classification and localization. Using the same CNN features computed for these tasks, we perform also robust artwork recognition. To improve the recognition accuracy we perform additional video processing using shape based filtering, artwork tracking and temporal filtering. The system has been deployed on a NVIDIA Jetson TK1 and a NVIDIA Shield Tablet K1, and tested in a real

world environment (Bargello Museum of Florence). [1]

10.1 Introduction

Digital and mobile technologies are becoming a key factor to enhance visitors' experiences during a museum visit, e.g. creating interactive and personalized visits. Personalization is viewed as a factor in enabling museums to change from "talking to the visitor" to "talking with the visitors", turning a monologue to a dialogue. This applies especially to audio guides since, similarly to a real museum guide, they must adapt their content to the needs and interests of the visitors [14]. Whether personalization addresses on-line exhibitions [14], on-site display of artworks [74], or both on-line and on-site [153], there is a need to obtain information about the behavior of the visitor, e.g. what he is looking at, for how long, and what other events happen during the visit. In this Chapter we address the problem of creating a smart audio guide that adapts to the actions and interests of the visitor of a museum, understanding both the context of the visit and what the visitor is looking at.

The goal of this work is to implement a real-time computer vision system that can run on wearable devices to perform object classification and artwork recognition, to improve the experience of a museum visit through the automatic detection of the behavior of users. Object classification, sensors and voice activity detection help to understand the context of the visit, e.g. differentiating when a visitor is talking with people or his sight is occluded by other visitors, e.g. understanding if he has friends that accompany him during the visit to the museum, or he is just wandering through the museum, or if he is looking at an exhibit that interests him.[2] Artwork recognition allows to provide multimedia insights of the observed item automatically or to create a user profile based on what artworks a user is looking at and for how long.

[1] This Chapter has been published as "Deep Artwork Detection and Retrieval for Automatic Context-Aware Audio Guides" in *ACM Transactions on Multimedia Computing Communications and Applications, 35, pp. 1-21, 2017* [131] and as "Portable computer vision for new intelligent audio guides" in *EVA 2017 Florence - Electroning Imaging & the Visual Arts, 2017* [13]

[2] `https://vimeo.com/187957085`

10.2 Related Work

Personalized museum experience.

The personalization of a museum visit may address the on-line experience in a virtual museum, the on-site experience in the museum itself, or both cases.

In [14] web personalization in museums is motivated by the advantages that it provides in improving usability of museum web sites and the facilitation of the learning process implied in a visit. Personalization is considered a new communication strategy that improves relationships between visitors and the institution. In [153] it is presented the Cultural Heritage Information Personalization (CHIP) system, that bridges on-line and on-site tour guides creating a personalized visit tour through a web site and then downloading the guide on a mobile device with RFID sensors that track the visitor in the museum. Tour information and the rating of artworks, if provided by users, are then sent back to the web site to update the user profile. Interactive digital guides have been used in [167] and [80] to analyze and predict the behavioral patterns of museum visitors, according to four main patterns that were initially identified through ethnographic observations by [37]. The works show that the four patterns can be identified using features such as average time spent on each artwork, percentage of observed artworks, etc. In [76] augmented reality (AR) on a mobile device is coupled with a personalized interactive storytelling experience, e.g. adapting the guide based on the age of the visitor, providing a gamified experience to children. In [74] a non-intrusive computer-vision system has been presented, based on person re-identification of museum visitors observed through surveillance cameras. The system identifies the artworks that are observed by museum visitors and measures how much time is spent looking at each artwork, to create a personalized user profile. At the end of the tour the user profile is used to create a personalized exploration of multimedia content on an interactive table, providing more information on the items that most attracted the visitor, and suggesting additional visits and tours.

Object detection and recognition

After the breakthrough of convolutional neural networks in image classification brought by Krizhevsky *et al.* [79], several works have used similar or derived strategies to solve other image and video related tasks [39,60,61,123].

A simple, yet dramatically effective strategy pioneered by Girshick *et al.* is to extract CNN features from regions of an image. Further improvements in localization and accuracy are obtained using a bounding box regressor and fine-tuning the CNN features on the detection task. The task of computing a full forward pass for every sub-window is extremely time expensive even for moderately shallow networks. More up-to-date works [60,123] avoid this burden by computing a single full resolution convolution on the whole frame and then performing classification and bounding box regression over a region of interest computed over the last convolutional layer. Fast R-CNN avoided the computation of multiple full forward passes, nonetheless it required expensive resources to compute object proposals, often generated with low-level features such as edges [172]. Ren *et al.* [123] removed this further computation bottleneck by learning a lightweight object proposal sharing the same features of the network used for object detection.

A more recent class of approaches tries to generate a set of class-labeled bounding boxes with a single pass of a convolutional network [87,121]. Redmon *et al.* argue that You should Only Look Once (YOLO) at frames, using an architecture inspired by Inception [138] focused on reducing the network size and the computation. The main idea is to produce, as an output, a tensor of size $N \times N \times |C| \times 5$, representing the coordinates and probabilities, for each of the C categories, for N^2 evaluated locations. Liu *et al.* proposed an approach named Single-Shot Detection (SSD), which is very similar to YOLO, but differs in the fact that it removes all fully connected layers allowing to predict bounding box using small convolutional filters on the last convolutional activation map. One advantage of SSD is that it allows to evaluate more windows, at multiple scales, by computing convolutions on previous output layers.

Object detection and recognition

After the breakthrough of convolutional neural networks in image classification brought by Krizhevsky *et al.* [79], several works have used similar or derived strategies to solve other image and video related tasks [39,60,61,123]. A simple, yet dramatically effective strategy pioneered by Girshick *et al.* is to extract CNN features from regions of an image. Further improvements in localization and accuracy are obtained using a bounding box regressor and fine-tuning the CNN features on the detection task. The task of computing a full forward pass for every sub-window is extremely time expensive even

for moderately shallow networks. More up-to-date works [60,123] avoid this burden by computing a single full resolution convolution on the whole frame and then performing classification and bounding box regression over a region of interest computed over the last convolutional layer. Fast R-CNN avoided the computation of multiple full forward passes, nonetheless it required expensive resources to compute object proposals, often generated with low-level features such as edges [172]. Ren *et al.* [123] removed this further computation bottleneck by learning a lightweight object proposal sharing the same features of the network used for object detection.

A more recent class of approaches tries to generate a set of class-labeled bounding boxes with a single pass of a convolutional network [87,121]. Redmon *et al.* argue that You should Only Look Once (YOLO) at frames, using an architecture inspired by Inception [138] focused on reducing the network size and the computation. The main idea is to produce, as an output, a tensor of size $N \times N \times |C| \times 5$, representing the coordinates and probabilities, for each of the C categories, for N^2 evaluated locations. Liu *et al.* proposed an approach named Single-Shot Detection (SSD), which is very similar to YOLO, but differs in the fact that it removes all fully connected layers allowing to predict bounding box using small convolutional filters on the last convolutional activation map. One advantage of SSD is that it allows to evaluate more windows, at multiple scales, by computing convolutions on previous output layers.

Object recognition on mobile devices

The availability of multi-core CPUs and GPUs on mobile devices has recently allowed to implement multimedia and computer vision methods on smartphones, with particular attention to convolutional neural networks.

In [164] an analysis of the best CNN architectures for mobile devices has been performed, evaluating the impact of using NEON SIMD instructions available on ARM CPUs and BLAS routines. The authors propose to use a Network-In-Network (NIN) architecture, where neuron weights are compressed with Product Quantization, to reduce the memory occupation of the CNN network. This solution has been employed to implement a mobile system for food recognition, presented in [139]. The problem of food recognition using mobile devices has been addressed also in [96], where different CNNs are used to segment food, estimate the 3D volume and classify food, so to provide an estimation of the calories; however only the CNN for food classification has been ported to a mobile device. Speed improvement and memory requirements reduction of CNN execution, for mobile devices, has been obtained in [160] through weights quantization of fully connected and convolutional layers, and applying an error correction technique to minimize the estimation error of each layer. In [67] a framework to execute deep learning algorithms on mobile devices has been presented. The framework uses OpenCL to exploit the GPUs. The framework addresses the problem of thread divergence in GPUs through data padding. In [82] has been presented a framework for GPU-accelerated CNNs on Android devices, that uses SIMD instruction on mobile GPUs, parallelizing some types of layers on GPUs and others, that are less computationally intensive, on CPUs. The framework has been released as open source.

Voice Activity Detection

Voice activity detection (VAD) is the process of detecting when humans are speaking in a given audio stream. It is essential to improve further processing like automatic speech recognition or saving bandwidth in audio coding or conference systems.

The first VAD system was first investigated in the fifties to be used on TASI systems [17]. Early approaches to this problem were based on heuristics and simple energy modeling, by thresholding or observing zero-crossing rate rules [158]. These methods work well in settings where no background noise is present. More recent methods address this limitation by employing autoregressive models and line spectral frequencies [101] to observe signal statistics in current frame and compare it with the estimated noise statistics with some decision rules. However, most of these conventional algorithms assume that noise statistics are stationary over long periods of time, more than those of speech. Given the extreme diversity and rapid changes of noise in different environments, they can't detect occasional presence of speech. The most recent class of approaches for VAD are that of data-driven methods, that avoid to make assumption over the noise distribution. They usually use a classifier trained to predict speech vs non-speech given some acoustic features [38, 98]. Anyway, their performance degrades when the background noise resembles that of speech. The state-of-the-art methods exploit long-span context features learned through the use of recurrent neural networks [34, 41, 150] to adapt the classification on the basis of the previous frames.

The method presented in this Chapter addresses the problem of creating a personalized on-site museum experience using a non-intrusive computer vision algorithm that can be executed on board of an audio guide. Unlike works such as [82] and [67], no special framework has been used, and the problem of computational costs has been addressed using: *i)* a CUDA implementation of a CNN running on NVIDIA portable GPUs, and *ii)* designing the algorithm to exploit the same features used for object detection, classification and retrieval. The problem of the scarcity of training data, highlighted in [114], has been solved applying fine tuning to a pre-trained CNN. Moreover, we exploit on-board sensors and recent recurrent neural networks for voice detection to further understand the context of the wearer, like its movements and its interactions with other people.

The remainder of the Chapter is organized as follows:in Sect. 11.3 we describe the overall system architecture and its sub-systems; in Sect. 10.4 we describe our efficient method for detecting objects and how we obtain a reliable artwork identification using tracking and retrieval. Sect. 10.5 outlines the context modeling module based on voice and sensor input processing. The full system, comprising also the Android App is described in Sect. 12.2. Finally in Sect. 11.4 and Sect. 10.8 we present quantitative results on our system together with an user experience evaluation, then drawing conclusions in Sect. 12.3.

10.3 The System

The system we propose comprises several components that work together to enable a smart experience. Fig. 10.1 shows an architectural diagram illustrating the main submodules of the system. From a higher level view of our system, two main sub-systems are identified, one responsible to recognize artworks, (providing *Artwork id*) and one to model the *User status*. They generate input signals for the *Playback Control* module which is responsible to play descriptions at appropriate time.

Our system senses the environment through three main channels: a camera, a microphone and movement sensors. The three sources are accessed through an Android App which is also responsible as a front-end of the whole system.

The camera is used to understand what the user is looking at. A computer vision system is responsible to detect objects (*Object Detector*) and recognize

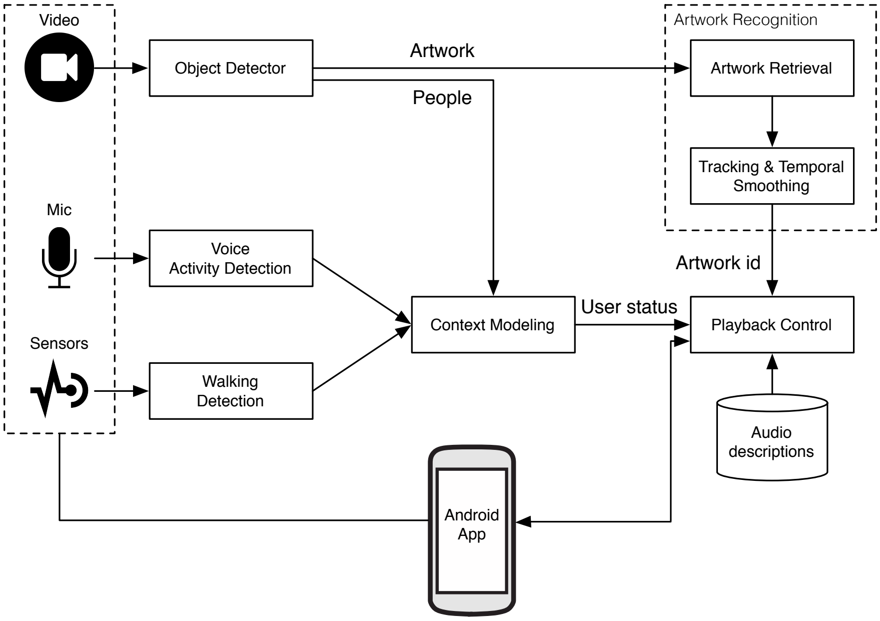


Figure 10.1: The overall system architecture.

what artwork the user is looking at (*Artwork Recognition*). Two sub-modules are highlighted in the recognition step: the first retrieves the most similar artwork from a database of known artworks and the second performs tracking to smooth out wrong predictions.

The *Context Modeling* module receives three behavioral signals: *People Detections*, *Voice Activity Detection*, and *Walking Detection*. These signals concur in generating a *User Status* signal. The microphone is used as a source for *Voice Activity Detection*, and movement sensors are necessary for *Walking Detection*.

10.4 Efficient Object Detection and Recognition

The smart audio guide we developed is based on an efficient computer vision pipeline that simultaneously performs artwork localization and recognition. The guide requires two main computer vision tasks to be solved: *i)* detection of relevant object categories: e.g. persons and artworks; and *ii)* for every detected artwork, reliable recognition of the specific artwork framed. Moreover, since we are dealing with a sequence of frames, in order to improve artwork recognition we take advantage from temporal coherence to make the output more stable.

Our system is based on YOLO [121], that is demonstrated to obtain accurate results even for moderate size networks. The main advantage of YOLO can be read in its acronym, i.e. it requires to look at the image only once. The process to generate scored boxes for each category of interest can be summarized as in the following. The whole image is split in 7×7 blocks. For each of the 49 regions a tensor of $5 \times 2 \times |C|$ is output. This tensor encodes two box predictions for each of the $|C|$ classes. Boxes are represented as a tuple $\langle x, y, h, w, s\rangle$. Non maximal suppression can be used to avoid multiple prediction for the same object. The confidence accounts for the accuracy of the bounding box and the probability of that class being present inside the given.

Differently from SSD [86], which is based on VGG-16, our YOLO-based classifier uses a much smaller network that allows the classifier to adhere with the memory requirements of an embedded system like the NVIDIA Tegra TK1 SoC. The architecture is derived from *Tiny Net*, a small CNN pre-trained on ImageNet, which allows the application to run at 10 FPS and fitting on the memory of a Shield Tablet.

The system network was fine-tuned to recognize artworks and people using our dataset. Recognizing people is relevant for two reasons: first we can exploit the presence of people in the field of view to create a better understanding of context, see Sect. 10.5; secondly, without learning a person model, it is hard to avoid false positives on people, since artwork training data contains statues, which may picture human figures. Learning jointly a person and an artwork model, the network features can be trained to discriminate between this two classes.

10.4.1 Artwork recognition

The rich features computed by the convolutional layers are exploited and re-used to compute an object descriptor for artwork recognition.

To ensure ease of deployment and update of the system, we base our artwork recognition system on a simple nearest neighbor step. We need to fulfill two important requirements: first our feature should be lightweight, i.e. low

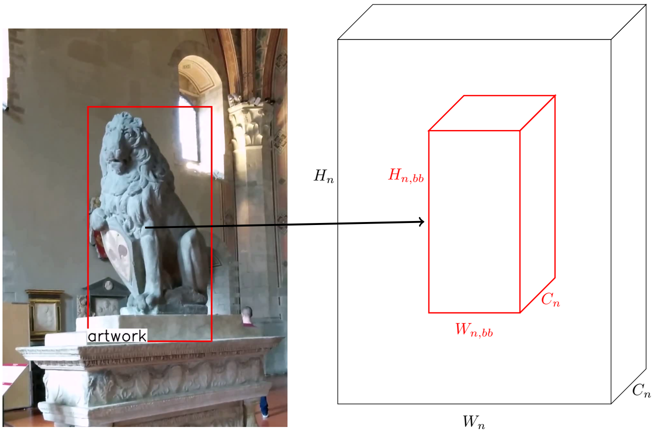


Figure 10.2: Feature extraction procedure for an artwork detection on a single convolutional feature map.

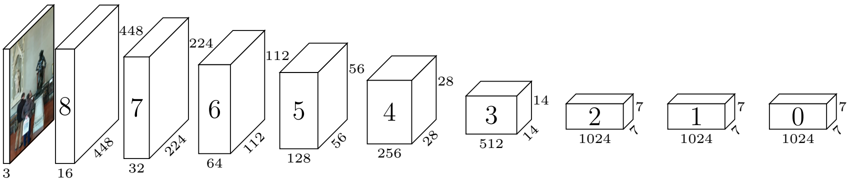


Figure 10.3: Our network architecture, with tensor size and layer numbering.

dimensional, in order to be stored on the device and reduce the computation time for feature comparison; second we must compute a discriminative representation for a region of the frame that may differ in size and aspect ratio.

To obtain a low dimensional fixed size descriptor of a region, we apply a global max-pooling over convolutional feature activation maps, as shown in Fig. 10.2. To increase the discriminative power, we concatenate such descriptor computed on two different feature maps. The region is remapped from the frame to the convolutional activation map with a simple similarity transformation.

Considering the activation map of the n^{th} convolutional layer, we have a tensor dimension of $W_n \times H_n \times C_n$. After the reprojection of the bounding box onto the feature map we end up with a smaller tensor with a size of $W_{n,bb} \times H_{n,bb} \times C_n$; $W_{n,bb}$ and $H_{n,bb}$ depend both on the network layer and the bounding box geometry, while C_n depends solely on the network layer and represents its number of channels. The max-pooling operation of the C_n channels over the $W_{n,bb} \times H_{n,bb}$ values generate a feature vector that is independent from the dimension of the bounding box.

Considering the architecture in Fig. 10.3 one could wonder which features are best to recognize the specific framed artwork, since leftmost layers have higher resolution and mostly represent the low-level structure of the image, while rightmost ones, are low resolution but encode higher level information, closer to the image semantics.

After an experimental evaluation, which is detailed in Sect.11.4, we selected, as combination, the features from layers 3 and 4, yielding a feature of size 768. The final bounding box descriptor is obtained by concatenation of the two max-pooled regions values and is L^2-normalized.

Considering a pre-acquired dataset of artwork patches $p_i \in \mathcal{D}$ and their artwork labels y, for each detected artwork d we predict a specific artwork label $y_{\hat{p}}$ finding the nearest neighbor patch

$$\hat{p} = \arg\max_i \langle p_i, d \rangle \tag{10.1}$$

The recognition system observes each frame independently and predicts artwork labels according to Eq. 10.1, this approach, in case of motion blur or quick lighting changes may produce incorrect recognition results. In the following we detail how we exploit temporal coherence to produce a more stable recognition output.

10.4.2 Artwork Tracking and Temporal Smoothing

High recognition accuracy is a requirement for the audio-guide, since mistaking an artwork for another may result in a bad user experience, e.g. this would result, at the interface level, in the audio guide presenting an artwork different from the one that is actually observed.

This is an extremely critical aspect and must be addressed, in order to improve the stability of the recognition system. We devise three strategies, based on the user location with respect to the artwork of interest and the continuous tracking of object bounding boxes.

To reduce the error rate our idea is to avoid performing artwork recognition on objects that may be too far from the user. Farther objects are unlikely to be of interest for the user, moreover the feature computed on a smaller bounding box has little discriminative power and likely leads to erroneous recognition.

Computing the actual metric distance from an artwork requires to perform real-time camera tracking and scene mapping. We believe that this accurate information is not required for our task and therefore we rely on a simple heuristic, comparing the areas of an artwork detection and the whole frame as in the following:

$$\frac{w_{bb}h_{bb}}{WH} > T \tag{10.2}$$

where WH is the frame area and w_{bb} and h_{bb} are bounding box width and height respectively, and T is a threshold (Fig. 10.4) empirically fixed. We name this strategy *Distance*. In our experiments we obtained the best results for $T = 0.1$, that as can be seen in Sec. 10.7.4, allows to reduce false recognitions by 50% w.r.t. not using the heuristic, at the cost of introducing a small number of missed recognitions.

Considering that there is continuity when the user walks around in the area, an artwork recognized frequently across a very short amount of time is probably the most correct. To exploit this, we continuously predict artwork labels as described in Sec. 10.4.1, but we consider a prediction only after it persists for M frames. We name this strategy *Consistency*. We implement it by tracking all artwork detection boxes with a greedy data association tracking-by-detection algorithm, requiring an IoU of consecutive bounding boxes of 50%. An example of this tracking is shown in Fig. 10.5.

With the same principle, 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. We call this strategy *Persistence*. We increment a counter p every time the recognition label for a box is unchanged, keeping track of the most frequent label $\overline{y}$. Every time a label $y*$ is different from $\overline{y}$ we decrement p. We predict the artwork identity as $y*$ only if $p > P > M$. This technique greatly reduces the number of false recognitions. In our experiments best results were obtained for $M = 15$ and $P = 20$.

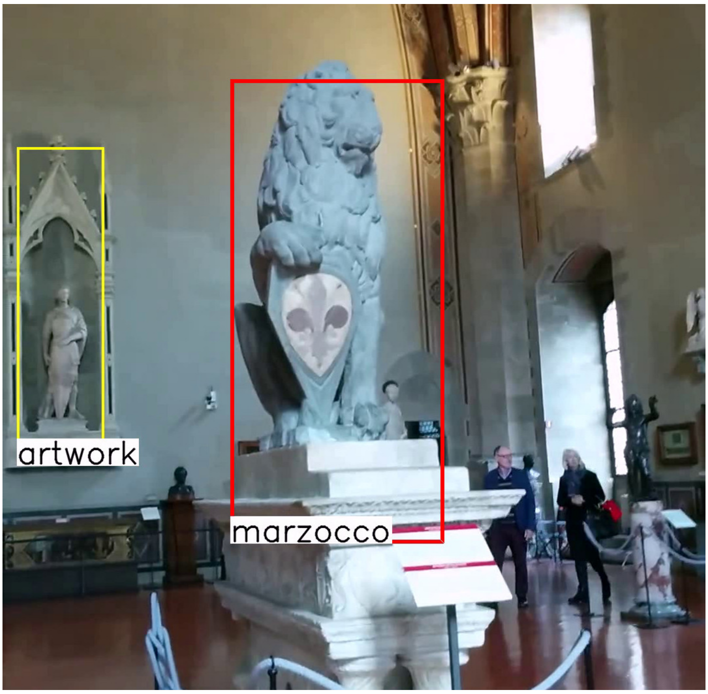


Figure 10.4: Shape based filtering: artwork in yellow (left) is not considered for recognition, not satisfying Eq. 10.2, while the other is recognized as "marzocco" (the heraldic lion symbol of Florence).

10.5 Context Modeling

To pursue the idea of an autonomous agent that is able to understand when it is the time to engage the user and when it should be inactive, it is essential to understand the context and the status of the wearer. In addition to the observation of the same scene the user is viewing through the wearable camera, we also try to understand if the user is busy following or participating in a conversation and if he is moving around the room, both independently from the visual data.

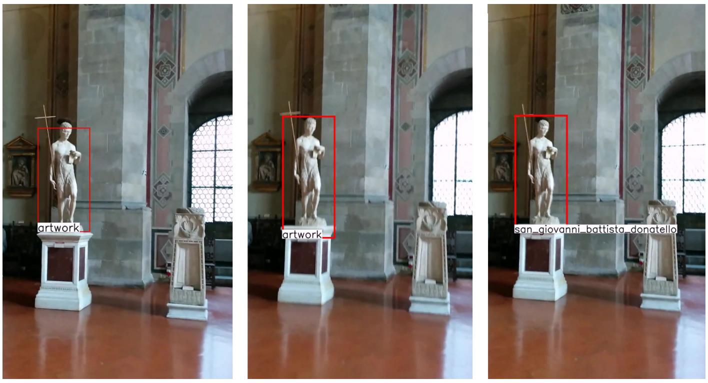


Figure 10.5: Example of artwork tracking, with $M = 15$. Only after a stable recognition over M frames the system labels the artwork.

10.5.1 Detecting conversations

Our audio-guide should be able to understand when the user is engaging in a conversation, if his field of view is occluded by other visitors or he is paying attention to another person or an human guide. In that event, it is reasonable to stop the audio-guide or temporarily interrupt the reproduction of any content, in order to let the user carry out his conversation. This should be of high priority and it should be performed even if the user is standing in front of an artwork. To identify this scenario, we use the device microphone to detect the presence of a nearby voice. We chose to employ a Voice Activity Detection (VAD) system for this task.

Typically, museums are mostly quiet environments where people tend to remain silent, to appreciate the artworks, and briefly talk between each other. Nonetheless, in some cases the environment can be noisy with the presence of music in background or some environmental noise. This requires the adoption of a VAD with automatic noise adaptation. The system will listen continuously to the environment, adapt to the local environment noise and detect when voice is present. Therefore, in order to run in real-time together with the computer vision module, it is essential to provide a lightweight system with low computational complexity. We adopted the system from [41], that is a state-of-the-art method based on a Long Short Term Memory recurrent neural network. This approach is able to model long range dependencies between the inputs (and thus accurately model environmental noise) and is highly scalable. The computational complexity for evaluating the networks is linear with respect to the number of input frames. Only a constant number of operations needs to be performed for every audio frame. We use the open source implementation and model available in the OpenSMILE framework [3].

Considering that a positive voice identification stops the playing of any description, it is imperative that the classifier has a low false positive rate. Unexpectedly stopping the reproduction due to a classifier error may result in a poor user experience. To this end, we evaluate an entire second of audio before emitting the prediction. We choose to use a classifier with a granularity of 0.01, so that, by exploiting all the predictions in this time frame, we can increase the stability of the prediction. The final prediction is the mean over the single classifications. We threshold this value according to the expected false positive rate, measured empirically on our dataset.

10.5.2 Sensors for walking detection

An important hint for understanding the context of the user is given by its movements. Standing still, walking or sitting can signal if the user is paying attention to some artwork or if he is uninterested in what he is looking at.

We make use of this information for mainly two purposes:

- If the user is walking fast then he is not probably interested in the visible artworks. This means that even if the visual system detects and recognizes an artwork, the audio description should not be started.
- If the user is standing still in front of an artwork and he is listening to the audio description, this should not be stopped, even if the visual system stops recognizing the artwork. This can happen mostly because of occlusions due to other people walking or standing between the visitor and the artwork.

To perform walking detection we use accelerometer data. We estimate the mean and standard deviation of acceleration magnitude from the training set. We subtract the mean from the acceleration magnitude and then we filter out peaks below the standard deviation. We consider each peak as a step. We then take into account a sliding window of 1 second, and consider the subject walking if at least a step is detected in the given window.

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

To detect if a person changes the facing direction, we estimate the orientation variation using gyroscope data. We average the orientation vector over the same 1 second sliding window. The facing direction is considered changed if the current orientation vector differs from the average for at least 45°.

10.6 System Implementation

The proposed system has been initially developed using a NVIDIA Jetson TK1 board, to test the performance of the vision system, introduced in Sec. 10.4, using a device designed for embedded systems. The board has a NVIDIA Kepler GPU with 192 CUDA cores, and an NVIDIA 4-plus-1 Quad-core ARM Cortex A15 CPU. Then the audio-guide application, named SeeForMe, has been deployed on an NVIDIA Shield Tablet K1 that has the same computational capabilities of the TK1 board, but it runs Android 5.0 instead of Linux, and it allows to develop a user friendly application that can support the visitor in his museum experience.

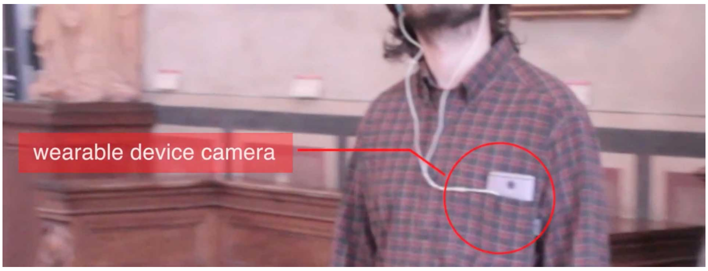
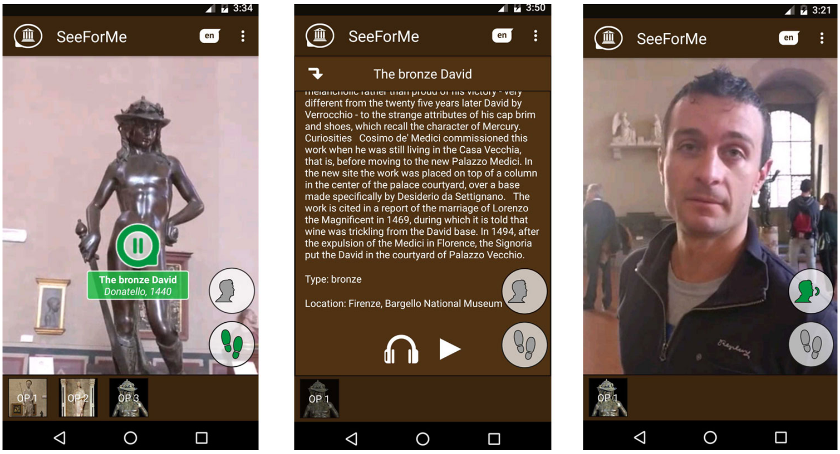


Figure 10.6: A visitor with the device camera in the pocket

We designed the application to handle three different user scenarios: *(i)* the user makes use of the application in a fully-automated way (placing the device in a front pocket with the camera facing forward, or hanging it on the chest using a special support), as shown in Fig. 10.6. In this scenario the system does not need any interaction and continuously observes the surroundings using the camera, choosing when to start and stop the audio by analyzing the user's behavior; in this modality the user can still interact with the application by using voice commands that are elaborated by the operating system and translated in the form of actions such as start/stop the audio; *(ii)* the user makes use of the application actively in a semi-automated way: after pointing the device towards the artworks the visitor is interested in, the system detects the artwork and provides the contextual audio guide, for which the audio can be started and stopped automatically or manually by the user; *(iii)* the user has completed the tour and wants to revisit his experience: to this end, the application provides a visual history of the tour represented as a carousel of artworks in temporal order. Through the carousel the user can select the artworks he visited, have multimedia insights and replay the audio guide.



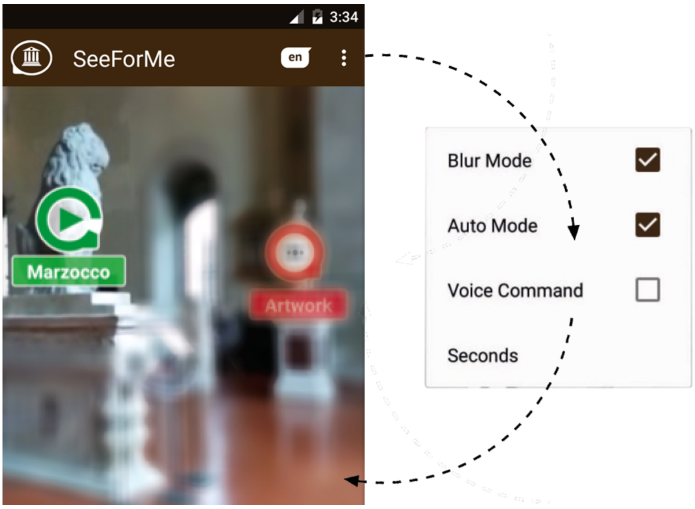
melancholic rather than proud of his victory - very different from the twenty five years later David by Verrocchio - to the strange attributes of his cap brim and shoes, which recall the character of Mercury. Curiosities Cosimo de' Medici commissioned this work when he was still living in the Casa Vecchia, that is, before moving to the new Palazzo Medici. In the new site the work was placed on top of a column in the center of the palace courtyard, over a base made specifically by Desiderio da Settignano. The work is cited in a report of the marriage of Lorenzo the Magnificent in 1469, during which it is told that wine was trickling from the David base. In 1494, after the expulsion of the Medici in Florence, the Signoria put the David in the courtyard of Palazzo Vecchio.

Type: bronze

Location: Firenze, Bargello National Museum

Figure 10.7: *(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,

Several application properties and modalities can be configured in the mobile app guide through a contextual menus reachable from the top right corner of the navigation bar. In Fig. 10.8 it is shown the contextual menus where there are two main modes: *i)* Blur mode, *ii) Auto mode*. The first one enables an app feature which blur the background of the artwork being framed by the visitor in order to focus his attention on the target. The Audio mode instead activates the automatic mode for the control of the guide audio stream. Voice Commands and interaction can also be enabled and disabled. Finally, a range in seconds can be defined to set a custom temporal window between the instant that the system recognizes the artwork and the start of



Blur Mode ☑
Auto Mode ☑
Voice Command □
Seconds

Figure 10.8: The contextual menus to configure the app properties. Here it is shown the appearance of the interface in blur mode.

the audio guide reproduction (these delay is marked visually by the green line in the icon which animates until it closes the circle, as shown in Fig. 10.8).

The mobile app has been developed using the Android SDK. The interface follows the guidelines of material design proposed by Google [4]. SQLite is used to persist the information on the device local storage. Communication between the app and the YOLO module is carried out using Java Native Interface (JNI) which enables the Java code running in the Java Virtual Machine (JVM) to call and be called by native applications. Data-interchange is performed through JSON messages. In particular, the YOLO module communicates with the mobile app passing data related to the current frame of the camera stream. This data comprises detected and recognized artworks and persons, with the coordinates of their bounding box, and booleans indicating if external speech and user movements have been detected.

[4] https://material.google.com/

10.7 Experimental Results

10.7.1 Dataset

We collected a dataset from footage captured in the Bargello Museum in Florence. Bargello Museum hosts a variety of artworks, featuring a large hall (Donatello Hall) with several masterpieces from Donatello. We use this hall as our testing ground. The collected data serves two distinct purposes: train and evaluate the object detector described in Sect. 10.4, and evaluate the full artwork recognition system.

Artwork imagery has been collected in a diverse set of illumination and viewpoint conditions. In fact, the Donatello Hall is an extremely challenging environment featuring a high ceiling with large glass windows. Therefore depending on the time of the day and the weather condition, artwork appearance may change significantly, because of light diffusion and camera sensor saturation. We collect $1,237$ images from all the statues in the Donatello Hall, in different lighting and viewpoint conditions.

For the object detection task we extract a subset of annotated images, splitting the data in training and testing. Artworks appearing in the training set do not appear in the testing set to correctly evaluate the performance of the detector. We added person images from PASCAL VOC2007 in order to have a more balanced training set. Fine-tuning our small network does not require a huge amount of data; we simply collected a balanced dataset of ~ 300 person and ~ 300 artwork images. We used vertical image flipping in training as data augmentation.

To evaluate the recognition system we annotated a larger set of images with the artwork id. To easily collect our recognition database we developed a tool based on our detection pipeline. We use our artwork detector to generate bounding boxes and the tracker described in Sect. 10.4.2, to link all boxes in a sequence, after a sufficient amount of frames of an artwork has been collected the user may simply select an existing id or enter a new record in the database. Considering the non-parametric nature of the recognition system discussed in Sect. 10.4.1, this process can be run multiple times to enrich the dataset.

Finally, to test our full pipeline, we use sequences accounting for $8,820$ frames. We also pay attention to include shots where multiple artworks are visible. In each frame, we annotated the bounding box and the label of each visible artwork. At the end of the process we collected a total of ~ 250

seconds of video with $7,956$ detections.

10.7.2 Artwork detection

In the first experiment, we evaluate the performance of the artwork detection system. After performing the fine-tuning of the network on our dataset, we run the trained detector on the test set and measure the average precision. As described in Sec. 10.4, we aim at detecting the artworks that are in front of the wearer and give less importance to the ones in the distance. As a result, we only consider detections of a minimum area T that are indicative of a small distance from the user. We report in Fig. 10.9 the average precision obtained by the detection system when varying the minimum area of the considered detections. The area is normalized with respect to the dimension of the video frame. It can be observed that the average precision increases with the minimum area of the box and reaches the maximum value of 0.9 at 40% of the area. This means that the classifier is more effective at recognizing nearer artworks. We note that increasing the minimum box size area is not always a guarantee that the detector will be more precise. While far detections are very prone to errors due to the small object scale, some detection errors may also be present at a near distance due to blur.

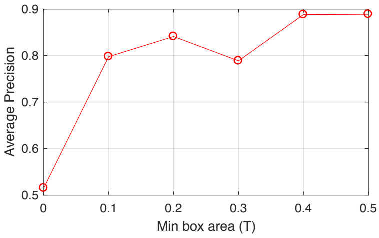


Figure 10.9: Average precision of artwork detection when varying the minimum box area.

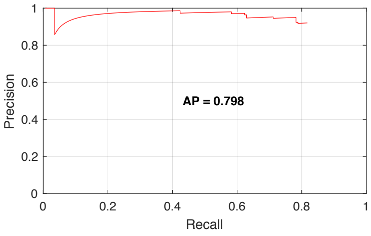


Figure 10.10: Precision-recall curve for artwork detection using a threshold $T = 0.1$.

Selecting a good value for the minimum box area is therefore a trade-off between a good precision and the proximity that a wearer has to be to an artwork. We chose the final value of $T = 0.1$, that provides a significant improvement of precision over the bare detector output and a maximum distance of ~ 5 meters. In Fig. 10.10 we show the precision-recall curve relative to the final T value. Our system has a very good precision at high recall rates. Hence, the curve exhibit only a small amount of loss in terms of precision until 0.8 recall. We note that higher recall can not be reached due to the T threshold selected according the results reported in Fig. 10.9. For this reason, the curve is truncated at that point.

10.7.3 Artwork recognition: nearest neighbour evaluation

In this experiment we evaluate the effect of the number of nearest neighbors on artwork recognition, in terms of precision. Descriptors are computed concatenating two layers of the network, according to the approach described in Sect. 10.4.1. Results are plotted in Figure 10.11 with accuracy using 1 nearest neighbour, where features extracted from layers 3 or 4 are combined with the other layers. The figure shows again that the combination of the 3^{rd} and 4^{th} layers provides the best results. In Figure 10.12 we report the accuracy when varying the number of nearest neighbours using the just selected best combination. We observe that 1 nearest neighbor provides the best performance in recognizing an artwork. Accuracy degrades when more nearest neighbours are used in voting the correct artwork id. This is due to the fact that the environment we are testing the system in, has high variability in lighting conditions. Moreover we acquired multiple poses for each artwork. It is clear that for each query only a few samples will be in the similar pose/lighting conditions while increasing the amount of neighbours will just add noisy data to the vote pool.

10.7.4 Temporal Processing Evaluation

In order to measure the effectiveness of the three strategies for temporal processing described in Sec. 10.4.2, we perform an experiment where several of their combinations are tested. The annotated video sequences are thus fed to a simulation of the system, where each combination of output bounding box and label is tracked and compared to the ground truth data. The thresholds are fixed at $T = 0.1$, $M = 15$ and $P = 20$. We measure the number of detections where the artworks are correctly and incorrectly labeled, and the number of times the system chose to output the “generic” artwork label.

We report in Table 10.1 the result of the evaluation. In the first test (T1), we measure the performance of the system without any additional criterion as baseline, i.e. the frame by frame output of the recognition system.

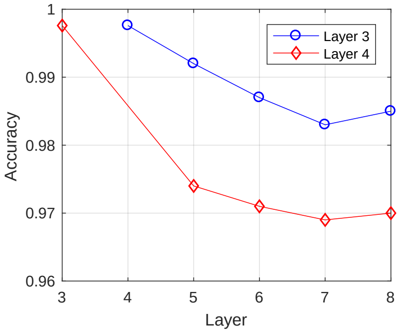


Figure 10.11: Recognition accuracy of combinations of layer 3 and 4 with layers $[3,\ldots,8]$.

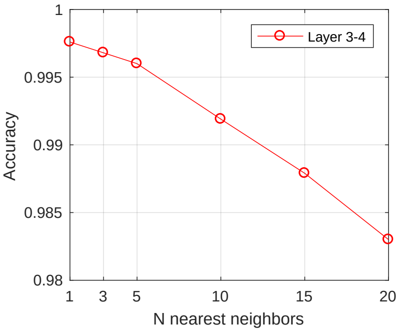


Figure 10.12: Recognition accuracy of the best layer combination (layers 3-4), varying the number of nearest neighbors.

We observe that the system outputs the correct artwork for the majority of the detections ($\sim$ 70%), while about 30% were labeled as an incorrect artwork. By adding the *Distance* criterion, we see in test T2 that a slightly lower amount of detections were correctly labeled, but about half of the incorrect recognitions were considered generic, instead. This confirms that a large amount of errors are made on farthest artworks, since they are more difficult to recognize. In test T3, we observe the outcome of the *Consistency* strategy. In this case, almost all the incorrect artwork recognition are successfully exposed and classified as generic artwork output. This is due to the uncertainty of the vision system that rapidly shifts its prediction from frame to frame. In test T4, as seen in T2, adding the *Distance* criterion to *Consistency*, reduces the Incorrect recognitions. While the *Consistency* criterion by itself is able to almost nullify the incorrect recognitions, it is not robust to sparse errors. In fact, the system often swings from the correct recognitions to the generic label. This issue is resolved when combining this stringent strategy with the *Persistence* one, in test T5. This is visible quantitatively in the gain of the number of correct recognitions and the relative decrease of generic outputs, at the expense of increasing the incorrect ones. Combining all the criteria, as in T6, leads to a very low number of incorrect detections and a reasonable number of neutral artwork outputs, confirming our intuition about the efficacy of the three strategies. With only 22 wrong detections, the system predicts a wrong label approximately less than one

Table 10.1: **Performance by applying the three strategies for temporal smoothing:** C stands for Consistency, D for Distance and P for Persistence. We report the number of detections where, respectively, the artwork was correctly recognized, the artwork was misplaced for another one and where the system chose to output a generic "artwork" label.

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%)

cumulative second every ~ 5 minutes of video.

10.7.5 Voice Detection Evaluation

In this experiment we test the performance of the voice activity detection system on our dataset. We consider two simple strategies to emit a classification per second, namely Sample and Mean. The Sample strategy is just evaluating the classifier on a single audio frame per second, sampled at the beginning of a new second. This has the advantage to require only a single evaluation of the net. The Mean strategy, instead, consider all the predictions of net in a second and finally emits the mean of the values. This is more robust to the fluctuations of the classifier, at the expense of running the net continuously. With both strategy, in order to minimize the number of false positives, we measure the performance of the classifier varying the positive threshold.

We report the receiver operating characteristic (ROC) curve of the two strategies in Fig. 10.13. We observe that both strategies have a high area under the curve (AUC), meaning that they correctly predict the presence of the voice most of the time. The Mean strategy has a higher AUC and has always an higher true positive rate at the same false positive rate than Sample. This confirms that the Mean strategy is more robust than Sample.

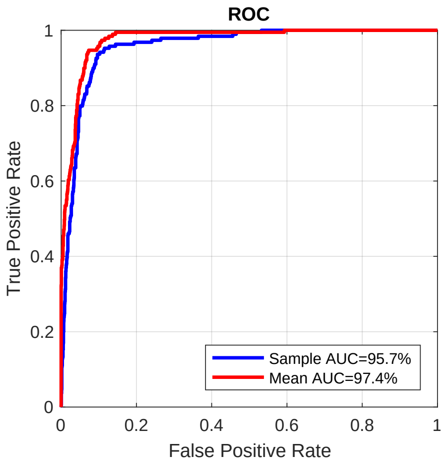


Figure 10.13: Receiver operating characteristic curve of the tested voice activity classifiers.

10.8 User experience evaluation

Modern tourist guides have their origins dating up to 17^{th} and 18^{th} centuries *Grand Tour*, and their role has become a key component in modern tourism experiences and applications. Guide functions can be highly specialized and require a lot of expertise and interpersonal skills to satisfy tourist needs. [28] describes guide roles as characterized by instrumental (guide), social (animator), interactional (leader) and communicative (intermediator) functions. Instrumental functions represent services capable to convey essential tourism information such as path finder to artworks location and related infos. Interactional features offer the ability to create a relation between the user and the contextual environment (e.g. informations about artworks). Improving this ability also means improving interaction. Sociality involves all the activities aiming at engaging the users with collaborative and not isolated experiences. Communicative functions facilitate access to artwork insights and targeted content, e.g. pointing out objects of interest. All these functionalities are fulfilled at their best by humans and modern audio guides have only partially replaced the complex role of the human guide. On the other hand the use of technology has improved aspects such as efficiency, sociality and autonomy in providing information communication under the so-called smart tourism

Table 10.2: **Functions comparison of our guide with respect to human and traditional audio guides**

Type	Instrumental	Social	Interactional	Communicative
Human Guide	∗∗∗	∗∗∗	∗∗∗	∗∗∗
Audio Guide	∗	−	−	∗∗
SeeForMe	∗∗	∗∗	∗∗	∗∗

paradigm. In Table 10.2 we compare guide-role functions as provided by human and traditional audio-guides with those available in our system.

The main differences between traditional audio guides and our system can be found in the interactional and social aspects of the provided experience. In traditional audio-guides the fruition of content is for the most part passive and the user has a low control on the reproduction of content. As regard to this, SeeForMe offers a more user friendly experience giving the possibility to interrupt the audio playback manually and automatically, and to restart the reproduction from the last point. Playback control can be achieved also using voice commands. Furthermore, activation of contents in audio-guides is cumbersome: locations or room numbers have to be searched and inserted manually reducing usability whilst SeeForMe allows automatic artwork recognition; this fact results in further differentiation of the Instrumental function, even in case of automatically triggered guides (e.g. those using RFID): an audioguide, being completely passive can not direct the visitor, while SeeForMe, highlighting the presence of other artworks as shown in Fig. 10.8, can direct the visitor within the museum. As for sociality, if it is true that social networking mechanisms are commonly provided in tourism apps for mobile phones, these functionalities are intended for virtual or remote users and not real companions. Indeed, audio-guides hinder communication between visitors (especially group visitors) and make people feel isolated, causing them to stop using devices and applications in order to join others. SeeForMe in this sense is more social because it automatically understands the context detecting if the user loses attention or simply is speaking with someone else, adapting the interaction with the system consequently.

In order to assess the whole experience offered by the system in a real environment, we conducted an evaluation of its usability. According to ISO, usability is defined as “the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use”. However, there are several usability models and types of assessments, like ISO standards on quality models (ISO 9126), user-centered design (ISO 9241) or user-centered approaches. A review of techniques for mobile application usability evaluation is provided in [104].

The usability study was performed with the popular Standard Usability Scale (SUS) [15], that follows a user-centered approach. Testing a user interface with SUS means, given a scenario of use and one or more tasks to solve, administer a 10 point questionnaire to a group of users. SUS is a Likert scale [143], therefore questions address extreme cases, with opposite meaning and alternating positive with negative sentences. Answers to questions are numbers from 1 to 5, expressing all ranges from “Strongly Disagree” to “Strongly Agree”. This testing strategy has been proved effective in removing acquiescence bias. The alternation of positive and negative items makes sure that users read it carefully. Nielsen states that it is sufficient to collect 5 polls to find the 85% of design errors of an interface or experience [107]; in these tests we recruited twelve persons, divided in two groups of six people each.

We tested two different scenarios, supervised and unsupervised, in which users were asked to perform two simple tasks: *i*) “Activate the audio-guide for one or more artworks of your interest”, *ii*) “After the visit, use the app to find again the information about one artwork you have seen”. In the former a group of people receives a spoken detailed description of our system, thoroughly explaining the Android app functionalities and also detailing insights on the recognition engine. In the latter scenario instead, users are given the same two simple tasks but without any explanation on the application functions.

After normalization SUS scores are expressed in the range $[0-100]$. They do not represent percentages but can be interpreted with an adjective rating [6]. A score over 68 means that the user interface or experience design is above average [128] and that tasks can be completed without too much fatigue. Scores above 80 usually means that the interface is correctly designed and that the user experience is enjoyable.

We obtained an average SUS of 74.0 for the unsupervised scenario and 79.5 for the supervised scenario. The small gap in scores measured in the two scenarios, and their closeness to 80, means that the user interface is easy to use and that the training provided by expert users is not strictly required to perform tasks correctly. Nonetheless, considering that the user experience increased when users received a brief tutorial on the features and technical details, means that there is some room for further improving the design of interface and user experience of our app.

Users, when interviewed, mostly agreed that the automatic start/stop of the guide is the feature that makes the experience smooth. Regarding negative aspects of our system, most of the points made by users were about the need to access menus to change the language or other options.

10.9 Conclusion

We have presented a system running on the NVIDIA Jetson TK1 and on NVIDIA Shield Tablet K1. Our approach jointly solves two problems: contextual analysis and object recognition. We apply our efficient video processing pipeline and multi-sensor analysis to improve museum experience. Our method allows to profile in real-time visitor interests and to provide instantaneous feedback on the artworks of interest. We exploit audio and sensor data to improve the user experience reducing the intrusiveness of the smart audioguide.

Our Android app, allows users to switch between a fully automated experience to a more interactive mode. Moreover, after a visit is completed it is possible to for the user to look back and listen, or read, again about the artwork that gathered his interest.

Usability testing revealed few pitfalls of our experience design, but users where satisfied on average and provided some suggestions to improve the user interface further.

Chapter 11

Locomotion by Natural Gestures for Immersive Virtual Environments

In this Chapter we evaluate methods to move 'naturally' in an Immersive Virtual Environment (IVE) visualised through an Head Mounted Display (HMD). Natural interaction is provided through gesture recognition on depth sensors' data. Gestural input solutions in the literature to provide locomotion are discussed. Two new methods for locomotion are proposed, implemented in a framework used for comparative evaluation. Perceived naturalness and effectiveness of locomotion methods are assessed through qualitative and quantitative measures. Extensive tests are conducted on the locomotion considering also: 1) obstacles in navigation; 2) interaction with virtual objects during locomotion. This is done with the aim to identify methods capable to provide a full body experience in an IVE. Results show that one of the methods for locomotion we propose has a performance comparable to established techniques in literature. Outcomes may be exploited to improve the naturalness of users' movements in IVEs and help to unlock new strategies in providing IVEs for learning, training, collaboration and entertainment, also with respect to users with

disabilities. [1]

11.1 Introduction and Related Work

Effective IVEs require intuitive interfaces controlled in a way that resembles real world experiences [144]. 'Incompatible spaces' is a common issue that researchers in HCI have to face when providing natural interaction in IVEs. In fact, IVEs allow free movement and infinite walking but the physical environment where the simulation is taking place presents spatial constraints.

There are several solutions in the literature to allow infinite walking in IVEs, still maintaining in users a realistic sensation of walking. These solutions can be classified in four groups which exploit:

a) *Additional Hardware*: unidirectional and omnidirectional treadmills, footpads and rotating spheres have been used to simulate natural walking maintaining fixed the user position in the environment [68,134]. These approaches are not easy to set up, require to secure users, are cumbersome and costly. Furthermore Natural User Interfaces (NUIs) do not contemplate the mediation of physical devices as controllers;

b) *Redirected Walking*: a set of reorientation and repositioning techniques which exploit virtual *stimuli* [136], e.g. giving the impression of walking straight to users moving in a circle [120] or using procedural layout generation [149]. Although these methods provide a good sense of presence, obstacles and physical constraints of the environment are still an issue;

c) *Software-based navigation*: interfaces featuring positional tracking supported by navigation tools. In [18] the tracking area, visualised as a 'magic carpet', can be repositioned using an appropriate tool for long-distance navigation. In [26] positional tracking is used in a restricted walking space whose physical boundaries, displayed in the IVE as a barrier tape, can be moved with a joystick. These solutions are not fully natural and require from users additional cognitive efforts while moving;

d) *Gesture Recognition using cameras*: vision-based methods for locomotion recognition have the advantage of not requiring additional hardware as interface controller. They solve several issues with respect to the solutions in a) and b), i.e. infinite walking and space constraints. However, it is difficult

[1] The work presented in this Chapter has been published as "Locomotion by Natural Gestures for Immersive Virtual Environments" in *Proceedings of the 1st International Workshop on Multimedia Alternate Realities (AltMM '16), 2016* [51].

to design and agree on a natural gesture to move. Furthermore, fatigue can affect the use of gesture-controlled interfaces, especially when reproducing a continuous action such as walking.

Methods of locomotion proposed in this Chapter fall within solutions in d). These are the most appropriate for NUIs mimicking real world interactions without the need of specific controllers. Several gestures have been defined in the literature which allow infinite walking in IVEs. Walking-In-Place (WIP) is the most common interaction paradigm: users can move in the IVE while remaining stationary [43, 120, 155]. Although WIP is usually referred as a form of compensating locomotion, the gesture is less frustrating for users than natural locomotion. Users moving naturally should repeatedly go forward and backward due to physical space constraints [108].

The Shake-Your-Head gesture in [142] allows the user to interact with the interface through head oscillations (i.e. as a transposition of the head movements observable in natural walking). Unlike the WIP technique, the user can both stand or sit in front of the interface.

This solves the fatigue problem caused by both standing and walking. Arm-Swing is a gesture performed oscillating the arms alternatively along the hips by a person as it is observed in natural walking. There's no implementation in the literature of a specific recogniser for Arm-Swing but the gesture is ranked second in the user study conducted in [108] where participants were given complete freedom in choosing gestures to complete tasks in a videogame. Free hand interactions have also been proposed and evaluated in literature to support locomotion in IVEs [19] as a mean to determine the direction of the movement.

The Chapter is organised as follows: in Sec. 11.2 we discuss the locomotion methods proposed and used in the evaluation; in Sec. 11.3 the framework and the input/output devices are presented; results, assessed through qualitative and quantitative experiments, are shown in Sec. 11.4.

11.2 Natural Interactions

Defining gestures in 3D IVEs exploiting natural interaction is easier than in 2D interfaces for the higher expressiveness that can be obtained by users simply acting like they do in the real world. 'Guessability' studies exploiting user-centered design show that in this scenario users' gestures are dominantly physical (e.g. walking moving knees) and metaphorical (e.g. selecting objects through pointing) [108, 115].

Building upon these studies, we evaluate four gestures for locomotion in IVEs (see Fig.11.1). Among these gestures, two are derived from the literature whilst the two others are novel. Locomotion methods have been chosen considering: 1) if gestures have been validated in similar studies; 2) the naturalness of the gestures with respect to the real world.

WIP (Walk-In-Place) The user walks in a stationary position. It is the most used in the literature, validated through qualitative and quantitative studies [43, 77, 120, 155];
Swing (Arms Swing) The idea is to replicate the natural oscillations of the arms during locomotion. It is a gesture demonstrated being actually performed by users freely interacting with a IVE [108];
Tap We propose a metaphorical gesture [115] for locomotion consisting in a tap with the index finger in the direction the user wants to start walking. It is a gesture not so far from the real world: people commonly use the index finger to show a walking direction;
Push We propose a metaphorical gesture consisting in closing and opening the hand while translating the hand itself forward with respect to the user elbow. In the real world it is the typical gesture to control locomotion machines moving a lever.

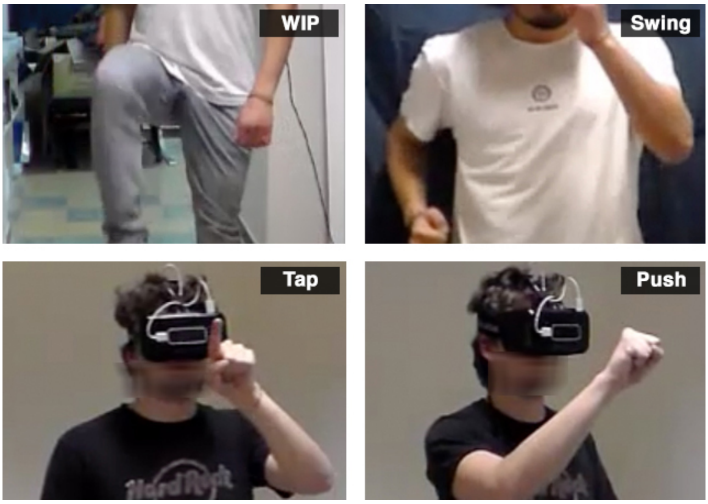


Figure 11.1: The four evaluated gestures for locomotion.

Shake-Your-Head gesture was not included in the study for two main reasons: 1) it can neither be classified as a natural gesture nor as a metaphorical one because the gesture has never been proposed by users in guessability studies; 2) it may cause motion sickness if used repeatedly in a HMD setup.

As regard to locomotion we must point out that the framework provides discrete and not continuous gestures in time. The reason is that users are aware of the fact that they are using methods of compensating locomotion and not natural locomotion. This is an essential feature for the usability of the IVE that otherwise: 1) it would strain too much the user with continuous activity (i.e. using WIP and Swing); 2) it would force the user to hold at least one of the hands always busy making it difficult to interact with virtual objects (i.e. using Tap). Once activated locomotion can be stopped with a 'Stop gesture' that the user can perform opening his hand in his field of view.

This gesture is motivated in [115] where it is demonstrated to be the preferred one by users performing a generic 'stop' action.

11.3 The Framework

The framework[2] consists in a library we developed that enables a first person controller to navigate and interact in IVEs created for the Unity3D engine[3] moving through the natural gestures described in Sec. 11.2.

Basic interaction with virtual objects is also made available.

The library allows to easily connect the interactive IVE with output and input devices, namely with an Head Mounted Display which visualises the 3D environment, and two tracking devices which provide the motion data gestures' detection relies on:

- *A Kinect v2*[4]. It tracks 25 body joint with millimetre accuracy and provides frame by frame data by which the WIP and the Swing gestures for locomotion are detected;
- *A Leap Motion*[5]. It tracks positions and rotations of each finger bone (24 per hand); mounted on the HMD facing in the user's field of view it is used to track hand movements and detect Tap and Push gestures for locomotion, Stop for interrupting locomotion, and gestures for interaction with virtual objects (i.e. pointing and grabbing).

[2] Demo video available at `https://vimeo.com/172710194/`

[3] https://unity3d.com/

[4] https://developer.microsoft.com/en-us/windows/kinect

[5] https://www.leapmotion.com/

For WIP and Swing gestures recognition we exploited the Microsoft Visual Gesture Builder NUI tool that generates gesture databases used to perform run-time detection through machine learning techniques (e.g. AdaBoost) applied to skeleton data. Leap Motion SDK instead provides Tap and Grab gestures recognition natively. For Push and Point gestures we have trained *ad hoc* classifiers. Looking direction equals walking direction in HMD for all the different gestures.

The library also includes UI components helpful to the user while exploring the IVE. Indicators of current direction and state of gesture recognition are superimposed on the 3D environment in order to give users proper awareness due to the absence of proprioceptive feedback. Furthermore, a virtual representation of user's hands is provided in the 3D environment to enhance sense of presence and ease virtual interactions.

11.4 Experimental results

An evaluation was conducted to determine how the proposed methods for locomotion in IVEs perform in terms of effectiveness and perceived naturalness. The four locomotion methods presented in Sec. 11.2 (i.e. WIP, Swing, Tap, Push) are evaluated comparatively, asking users to complete tasks of increasing difficulty.

Participants and procedure Evaluation was conducted with 19 participants (11 males and 8 females) aged between 21 and 39 years old (average 26.4, $\sigma = 5.8$). None of the participants had previous experience with IVEs or HMDs, but they reported a medium to high familiarity with technology (average of 4.4 on a 1 to 5 rating scale) and previous experience with first-person video games (average of 3.8 on a 1 to 5 rating scale). Locomotion methods and gestures for interactions were explained to all participants before the test. At the end of the session, participants were asked to fill a questionnaire.

Tasks and setting For the tests we created an IVE representing a forest. Two position in the virtual environment were defined by visual markers: a starting position A and a destination position B (see Fig.11.2).

Participants were asked to perform six tasks using all the four locomotion methods. In the easiest task users were asked to move from A to B.

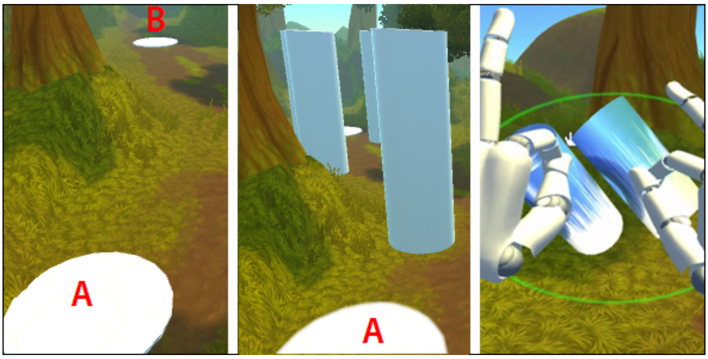


Figure 11.2: Task scenarios (i.e. T1, T2, T3).

Other tasks were defined combining further difficulties such as going back to position A, avoiding obstacles placed along the locomotion path and, at the same time, bringing an object from a position to another. Most of the cited works in the literature evaluate the naturalness of gestures for locomotion and interaction with virtual objects as separate topics [19,111,168] and, to the best of our knowledge, solving both problems together it is still an open issue which needs to be addressed in IVEs applications. For this reason, we introduced some tasks that contemplate the use of the Grab gesture to relocate a virtual object in the environment. The following tasks were defined:

T1 Move from position A to position B.
T2 Move from position A to position B and back to A.
T3 Move from position A to position B, avoiding obstacles on the path.
T4 Move from position A to position B and then back to A, avoiding obstacles on the path.
T5 Move from position A to position B, grab an object and then bring it back to A.
T6 Move from position A to position B, grab an object and then bring it back to A, avoiding obstacles on the path.

The order of the used locomotion methods was randomised so to eliminate potential order-related bias. Since the Swing and Grab gestures are incompatible (i.e. Swing assumes that both arms are occupied), results of T5 and T6 are n.a.

Measures Locomotion techniques in the framework were evaluated using both qualitative and quantitative methods. Naturalness and effectiveness of locomotion were assessed using the following measures:

Perceived Naturalness. Following the heuristic evaluation method for natural engagement in IVEs proposed in [137], we provided a questionnaire to collect subjective measures of naturalness of locomotion gestures from the participants, expressed on a 1 to 7 scale.

Overall preference. At the end of each session, we asked users to indicate which method they preferred.

Time Completion. A quantitative measure of the time required to complete each task. We did not define a maximum execution time and all users were able to complete all the tasks.

Collision Avoidance. This measure was proposed in [88] as a meaningful way to evaluate locomotion in IVEs. In two of the tasks including obstacles (i.e. T4 and T6) we counted the number of collisions occurred.

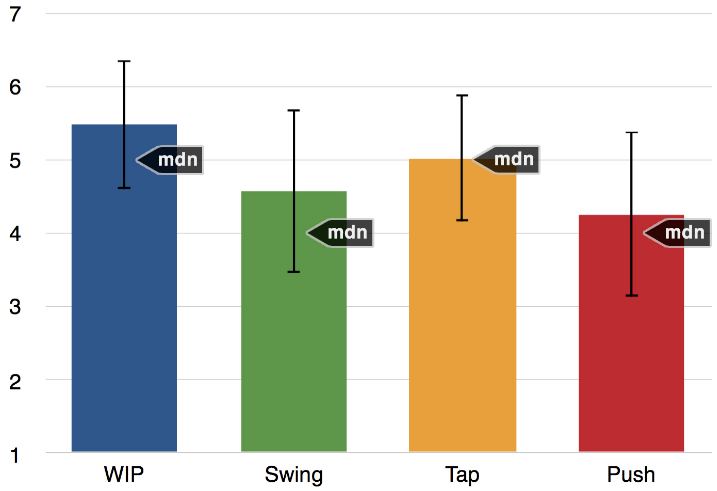


Figure 11.3: Perceived Naturalness of locomotion methods. The higher the better. The black bars stand for the standard deviation.

Results Qualitative and quantitative results were statistically analysed to obtain a comparative evaluation of the four locomotion methods. Qualitative comparison in terms of *Perceived Naturalness* is shown in Fig. 11.3. All methods have a good rating, but highest scores were obtained by WIP (avg 5.47, mdn 5) and Tap (avg 5.02, mdn 5). Results of the *Time Completion* (see Table 11.1) and *Collision Avoidance* (see Table 11.2) tests reveal sensible differences between methods in terms of effectiveness. WIP and Tap methods result to be the fastest and less prone to collisions in almost every task. Tap in particular performs better than other methods in T5 (Table 11.1) and in T4 and T6 (Table 11.2). The gesture seems to overcome WIP in tasks that contemplate hand-based interaction (i.e. Grab) and *Collision Avoidance*. An explanation could be given by the verbal considerations of some testers that reported WIP to require a sort of bilateral integration between hands and legs. Results from the *Overall preference* questionnaire indicate that more than half of testers (10 out of 19) would choose Tap as locomotion method, while the remaining preferences were for Push (6 out of 19) and WIP (3 out of 19). The outcomes of the evaluation suggest that even though WIP is by far the most used locomotion technique in IVEs, novel gestures such as Tap could be adopted with comparable results in terms of effectiveness of user experience. Results in Tables 11.1 and 11.2 are preliminary: analysis of variance for statistical significance of means between groups of testers are needed and will be the subject of future work.

Table 11.1: Time Completion in seconds. The lower the better.

	WIP		**Swing**		**Tap**		**Push**	
	Avg	σ	Avg	σ	Avg	σ	Avg	σ
T1	**15**	2	**15**	2	16	3	21	6
T2	40	7	39	6	**30**	2	35	5
T3	**17**	2	19	7	20	5	26	9
T4	43	10	42	10	**37**	6	52	16
T5	56	13	n.a.	n.a.	**51**	9	64	21
T6	**53**	21	n.a.	n.a.	57	23	75	34

11.5 Conclusions

Providing IVEs' users with the best natural experience is a challenging task. Commonly IVEs are mediated by displays mounted on the head and there's a physical gap between real and virtual space. Infinite locomotion in virtual

Table 11.2: Collision Avoidance results showing number of collisions. The lower the better.

	WIP		**Swing**		**Tap**		**Push**	
	Avg	σ	Avg	σ	Avg	σ	Avg	σ
T4	0.46	0.78	0.54	0.77	**0.38**	0.61	0.84	0.98
T6	0.92	1.11	n.a	n.a	**0.61**	0.96	1.00	0.91

environments collides with the constraints of their fruition in spaces closed by walls or obstructed by obstacles. Natural interaction provides a solution to these issues through gesture recognition. We identify and comparatively evaluate four methods of locomotion (i.e. WIP, Tap, Swing, Push).

Qualitative and quantitative experiments are conducted through user testing.

Results show that two of the four methods perform better than the others (i.e. WIP and Tap) and that the Tap gesture we propose has similar and in some tasks better performance than the well established WIP locomotion technique. This evidence may be useful to researchers and interaction experts for designing IVEs and for providing whole body natural experiences. Furthermore, performance of Tap suggests that hand-based gestures for locomotion deserve further investigation. Although being metaphorical the Tap gesture was perceived as natural by testers. Its adoption could provide some advantages in certain scenarios: for example, it could be used in configurations with a seated user, resulting in a reduction of physical fatigue, and improve accessibility to IVEs even for users with reduced mobility.

Chapter 12

Natural Experiences in Museums through Virtual Reality and Voice Commands

In this Chapter we present a system for immersive experiences in museums using Voice Commands (VCs) and Virtual Reality (VR). The system has been specifically designed for use by people with motor disabilities. Natural interaction is provided through Automatic Speech Recognition (ASR) and allows to experience VR environments wearing an Head Mounted Display (HMD), i.e. the Oculus Rift. Insights gathered during the implementation and results from an initial usability evaluation are reported. [1]

12.1 Introduction

Nowadays, personalized mobile museum guides, augmented reality systems featuring see-through technology and HMD VR systems are the most popular trends for providing visitors with rich context-aware information in cultural heritage apps [146]. However, such technologies pose limitations to users with motor disabilities as they assume the ability to hold a device or to move and interact with the surrounding real or virtual Immersive Museum

[1] Part of this Chapter has been published as "Natural Experiences in Museums through Virtual Reality and Voice Commands" in *Proceedings of the 23rd ACM international conference on Multimedia, 2017* [45].

Environment (IME) through controllers or natural gestures [52]. In latest years there has been a significant raise of voice interaction in games and 'serious games'. This is due to the proliferation of consumer devices with built-in capabilities for Automatic Speech Recognition (ASR) such as the Microsoft Kinect as well as to improvements of these systems in terms of recognition rates. However, though voice interaction has long been of interest to HCI as perceived like a natural way of communicating with a computer, it has not yet freed itself from being regarded as a supplement to traditional controller-based or gestural input. In fact, there is still little research on how to exploit progress in ASR for developing effective and accessible speech controlled interfaces [102].Nevertheless, some examples exist of humanoid conversational agents in Museum applications, but dialogue is poorly supported [12] and, also in advanced immersive solutions which exploit HMD, is restricted to few words [92]. In this regard, there are still significant issues related to the use of VCs in 'games' and interactive exhibits that can be summarized as follows: 1) perceived distance between the player and the game character defined as 'identity dissonance' in [21]; 2) the social context where voice interaction takes place (e.g. the quiet environment of museums, privacy concerns); 3) errors in ASR (due to noise, spelling, etc.); 4) restricted freedom of speech in limited domain applications with VCs constituted by simple words or short phrases due to the difficulty of ASR in the wild.

In this demo we propose some ideas on how to alleviate these issues experiencing an IME displayed through an HMD and made walkable using VCs. The system was conceived as a natural interface for users with motor disabilities, so that they can visit a museum not only remotely but also exploiting VCs exclusively. The player can navigate the museum and obtain information through Voice Commands to a Virtual Museum Guide agent (VMG). Commands have a certain degree of freedom since are automatically fed and augmented *via* a semantic storage provided with a reasoner capable of inferring concepts.

12.2 The system

The system[2] is composed by three main modules, implemented in a library for Unity 3D[3], respectively in charge of: 1) importing and setting up the

[2] Demo video available at `https://vimeo.com/miccunifi/museum-voice-commands`

[3] `https://unity3d.com`

IME; 2) performing ASR, augmenting and detecting Voice Commands; 3) allowing interaction and navigation in the environment.

Setting up the Environment The library allows to insert and place artworks (paintings and sculptures) in a 3D Museum model using Unity scripts, that can be attached to 3D objects. Artworks can be described using triples $\{s, p, o\}$ through ontologies imported in or created by the system (e.g. specifying image URIs, authors and artistic movement artists belong to). Possible questions can be defined as instances of the *Question* class through a script attached to the First Person Controller. Multiple ontologies can be used and extended creating new classes, instances and properties which support both literals and resources. For parsing and managing ontologies and triples in Unity the system exploits the dotNetRDF Opensource Library[4]. Statements are saved in N-Triples format and then imported in the Apache TDB Jena semantic storage[5].

Speech Recognition Speech Recognition is performed by the System exploiting the Microsoft Speech API (SAPI) 5.3, the native API for Windows[6], and mapping VCs to a dynamic grammar using rules. This is done in order to allow the user to ask questions and express commands in the virtual space. Rules define patterns and word sequences to be matched against the vocal input. Rules are represented as a graph of states. States (or group of words) are part of a sentence which mark a particular part-of-speech in the context (they identify the relationship of a state with adjacent and related states in the sentence; e.g. subject, predicate and direct objects). Rules and patterns are described in an XML-format grammar that conforms to the Microsoft Speech Recognition Grammar Specification (SRGS) Version 1.0. The grammar contains variants of interrogative, exhortative and desiderative sentences and is dynamically created through SPARQL queries. In this way, questions and requests by the user in the domain are intended as voice commands by the natural interface. A predefined set of instances of a *vc:Question* and *vc:Request* classes has been provided with the library. Requests and questions have three default properties which are *vc:hasSubject*, *vc:hasPredicate* and *vc:hasDirectObject.* The instances of these classes *vc:hasSubject rdfs:range vc:Character*; *vc:hasPredicate rdfs:range*

[4] `http://www.dotnetrdf.org/`

[5] `https://jena.apache.org/`

[6] `http://bit.ly/2qNEnWF`

vc:Predicate and *vc:hasDirectObject rdfs:range vc:Artwork, vc:Artist, vc:ArtisticMovement.* Direct objects are resources retrieved dynamically form the semantic storage via SPARQL and added setting up the environment in the Unity 3D Editor (e.g. "I'd like to see 'The Scream' by Edvard Munch"). Inference is also provided by the system. For example, given that:

```
Class(vc:ActionPainter complete intersectionOf(vc:Artist
  restriction(vc:exponentOf someValuesFrom(a:ActionPainting))))
Class(vc:AbstractPainter complete intersectionOf(vc:Artist
  restriction(vc:exponentOf someValuesFrom (vc:AbstractArt))))
Class(vc:ActionPainting partial vc:AbstractArt)
```

The following class inference can be derived:

- an Action Painter is an exponent of the Action Painting;
- Action Painting is a type of Abstract Art;
- an Action Painter is an exponent of the Abstract Art, so must be an Abstract Painter.

When concepts are inferred, the grammar is updated and rules added so that the user may ask additional questions such as "Which types of abstract art are present in the museum?" or "Is Jackson Pollock an abstract painter?". The inference engine solves in part the issue n. 4 expressed in Sec. 12.1 allowing more flexibility in questions and commands.

Interaction and Visualization The virtual museum is visualized through the Oculus Rift which provides immersion. The system has been designed to be used by people with motor disabilities in their own rooms or in a dedicated private space in this way excluding the social context of the interaction, and consequently embarrassment and privacy concerns related to VCs, and environmental noise (i.e. issue n. 2 and n. 3 in Sec.12.1). In order to increase naturalness, the user experiences the environment as a First Person Viewer. He is guided inside the museum by a VMG agent to whom he can ask questions using voice. In this scenario, the player has not to embody himself with a virtual representation. This mitigates the 'identity dissonance' issue (i.e. n. 1 in Sec. 12.1). Furthermore, verbal immediacy is demonstrated to have a significant impact on learning and sense of presence in IMEs [12, 62, 78]. To ease the user interaction, upon first access the agent lists the possible vocal questions the virtual visitor can ask. There are three default question instances in the semantic storage: 1) “Which artistic movements are displayed in the museum?”; 2) “What artists are there in the museum?” and 3) “What artworks are there in the museum?”. Additional questions, that dynamically populate the grammar for ASR, can be added manually through ontologies or inferred by the reasoner. Text-To-Speech (TTS) synthesis is used by the guide to explain possible questions and to give responses. Once the user has asked the question of interest, the ASR takes the audio stream as input and turns it into a text transcription. Acoustic models, lexicons and language models are used to search the best match of the input with the textual instances present in the grammar. Let’s say that the user ask question 1). The question is interpreted as a VC and mapped to a SPARQL query. Consequently, the guide will list all the pertinent information retrieved or inferred by the reasoner to the user using TTS. Then she will ask the user which artistic movement he is interested in. So the conversation can go on, and the user can make new requests (desiderative or imperative) to the agent who can satisfy them in two ways: 1) explaining the concept and asking new questions (e.g. listing all the artists of a particular movement) or 2) guiding the user to and describing an artwork of interest if he expresses the desire to know more about it (e.g. “I’d like to see ‘The Starry Night’ by Vincent Van Gogh”). In the latter case, the VMG guides the visitor to the place where the artwork is located walking through the halls of the museum. The idea is to give the user the natural impression of following behind a guide while she explains what she and the visitor are going to see. To make the guide move naturally through the museum environment avoiding obstacles (e.g. walls, sculptures) the A* algorithm is exploited. A* is an algorithm for path finding which can compute the shortest path between *vertices* in a graph. Given the 2D museum map, all the walkable surface and obstacles are mapped to a fine-grained grid modeled as a graph. The A* algorithm is able to find the least cost path from an initial node to a goal node. How the interaction between the player and the guide works is demonstrated in our demo video.

The usability of the system was preliminarily tested using the popular Standard Usability Scale (SUS) [16]. 10 users were asked to perform the task of navigating the museum using VCs obtaining insights from the VMG on at least an artistic movement and an artwork. Average SUS score was 71.0. Scores are in the range $[0-100]$ and over 68 mean that the interaction design is above average [129].

12.3 Future Work

ASR and language processing are going to be used in order to understand more complex types of phrases (not only factual, but also convergent, divergent, evaluative) [85]. Guide mouth's movements have to be made realistic and naturalness of TTS synthesis needs improvements. More accurate usability tests with users with motor disabilities should be conducted.

Chapter 13

Publications

This research activity has led to several publications in international journals and conferences. These are summarized below.[1]

Books

1. **A. Ferracani**, D. Pezzatini, L. Seidenari and A. Del Bimbo, "Natural Interaction in Medical Training. Tools and Applications", *SpringerBriefs in Human-Computer Interaction*, to appear, 2017

International Journals

1. L. Seidenari, C. Baecchi, T. Uricchio, **A. Ferracani**, M. Bertini, A. Del Bimbo "Deep Artwork Detection and Retrieval for Automatic Context-Aware Audio Guides", in *ACM Trans. Multimedia Comput. Commun. Appl.*, 35, 1-21, 2017.
2. S. Karaman, A. Bagdanov, L. Landucci, G. D'Amico, **A. Ferracani**, D. Pezzatini, A. Del Bimbo. "Personalized multimedia content delivery on an interactive table by passive observation of museum visitors", in *Multimedia Tools and Applications*, vol. 75-7, 2016.

International Conferences and Workshops

1. **A. Ferracani**, D. Pezzatini, L. Landucci, G. Becchi, A. Del Bimbo "Separating the Wheat from the Chaff: Events Detection in Twitter Data", in

[1] The author's bibliometric indices are the following: H-index = 7, total number of citations = 152 (source: Google Scholar on Month October, 2017).

Proceedings of the Content-based Multimedia Indexing International Workshop (CBMI 2017), Florence (IT), 2017.

2. A. Del Bimbo, M. Bertini, L. Seidenari, C. Baecchi, T. Uricchio, **A. Ferracani** “Portable computer vision for new ”intelligent” audio guides”, in *EVA 2017 Florence - Electroning Imaging & the Visual Arts*, Florence (IT), 2017.

3. **A. Ferracani**, L. Landucci, M. Faustino, G. X. Giannini, A. Del Bimbo. “Natural Experiences in Museums through Virtual Reality and Voice Commands”, in *Proceedings of the 25rd ACM international conference on Multimedia*, Silicon Valley, 2017.

4. F. Becattini, **A. Ferracani**, L. Landucci, D. Pezzatini, T. Uricchio, A. Del Bimbo. “Imaging Novecento. A Mobile App for Automatic Recognition of Artworks and Transfer of Artistic Styles”, in *Proceedings of Digital Heritage. Progress in Cultural Heritage: Documentation, Preservation, and Protection: 6th International Conference (EuroMed 2016)*, pp. 781-791, Nicosia, Cyprus, 2016. (**Best Paper Award**)

5. **A. Ferracani**, D. Pezzatini,J. Bianchini, G. Biscini, A. Del Bimbo. “Locomotion by Natural Gestures for Immersive Virtual Environments”, in *Proceedings of the 1st International Workshop on Multimedia Alternate Realities (AltMM ’16)*, Amsterdam (NL), 2016.

6. **A. Ferracani**, D. Pezzatini, M. Bertini, A. Del Bimbo. “Item-Based Video Recommendation: An Hybrid Approach considering Human Factors”, in *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval*, Firenze, 2016.

7. **A. Ferracani**, D. Pezzatini, R. Del Chiaro, F. Yang, M. Sanesi, A. Del Bimbo. “smArt: Open and Interactive Indoor Cultural Data”, in *Proceedings of the 23rd ACM International Conference on Multimedia*, Brisbane, Australia, 2015.

8. **A. Ferracani**, D. Pezzatini, M. Bertini, S. Meucci, A. Del Bimbo. “A system for video recommendation using visual saliency, crowdsourced and automatic annotations”, in *Proceedings of the 23rd ACM International Conference on Multimedia*, Brisbane, Australia, 2015.

9. **A. Ferracani**, L. Landucci, P. Pala. “Exploring 3D Virtual Environments through Optimised Spherical Panorama Navigation”, in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME 2015) - Demo Session*, 2015.

10. **A. Ferracani**, D. Pezzatini, A. Del Bimbo. “Roadie: Mobile Semantic Tourism Routes”, in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME 2015) - Demo Session*, 2015.

11. **A. Ferracani**, D. Pezzatini, A. Benericetti, M. Guiducci, A. Del Bimbo. "PITAGORA: Recommending Users and Local Experts in an Airport Social Network", in *Proceedings of the 23rd ACM International Conference on Multimedia*, Brisbane, Australia, 2015.
12. **A. Ferracani**, D. Pezzatini, A. Del Bimbo. "User Profiling for Urban Computing: Enriching Social Network Trace Data", in *Proceedings of the 3rd ACM Multimedia Workshop on Geotagging and Its Applications in Multimedia*, Orlando, Florida, 2014.

National Conferences

1. L. Seidenari, C. Baecchi, T. Uricchio, **A. Ferracani**, M. Bertini, A. Del Bimbo. "Object Recognition and Tracking for Smart Audio Guides", in *Proceedings of the 14th Italian Research Conference on Digital Libraries*, Udine, Italy, 2018. To appear.

Extended Abstracts

1. L. Seidenari, C. Baecchi, T. Uricchio, **A. Ferracani**, M. Bertini, A. Del Bimbo. "A Framework for Web-based Social Multimedia Search Engines", in *5th Workshop on Web-scale Vision and Social Media (VSM), ICCV 2017*, Venezia, Italy, 2017. (Published on the website `https://goo.gl/Fr47W2`)

Bibliography

[1] A. G. Anderson, C. P. Berg, D. P. Mossing, and B. A. Olshausen, "Deepmovie: Using optical flow and deep neural networks to stylize movies," *arXiv preprint arXiv:1605.08153*, 2016.

[2] R. M. Anwer, F. S. Khan, J. van de Weijer, and J. Laaksonen, "Combining holistic and part-based deep representations for computational painting categorization," in *Proceedings of the 2016 ACM ICMR*. ACM, 2016, pp. 339–342.

[3] Y. Arase, X. Xie, T. Hara, and S. Nishio, "Mining people's trips from large scale geo-tagged photos," in *Proceedings of the International Conference on Multimedia*, ser. MM '10. New York, NY, USA: ACM, 2010, pp. 133–142. [Online]. Available: http://doi.acm.org/10.1145/1873951.1873971

[4] D. A. at Al., "Google street view: Capturing the world at street level," *Computer - IEEE*, june 2010.

[5] R. Ballagas, M. Rohs, J. G. Sheridan, and J. Borchers, "Byod: Bring your own device," in *Proceedings of the Workshop on Ubiquitous Display Environments, Ubicomp*, vol. 2004, 2004.

[6] A. Bangor, P. Kortum, and J. Miller, "Determining what individual sus scores mean: Adding an adjective rating scale," *Journal of Usability Studies*, vol. 4, no. 3, pp. 114–123, 2009.

[7] J. Bao, Y. Zheng, D. Wilkie, and M. F. Mokbel, "A survey on recommendations in location-based social networks," *GeoInformatica*, November 2014. [Online]. Available: http://research.microsoft.com/apps/pubs/default.aspx?id=191797

[8] L. Baraldi, F. Paci, G. Serra, L. Benini, and R. Cucchiara, "Gesture recognition using wearable vision sensors to enhance visitors museum experiences," *IEEE Sensors Journal*, vol. 15, no. 5, pp. 2705–2714, 2015.

[9] F. Becattini, A. Ferracani, L. Landucci, D. Pezzatini, T. Uricchio, and A. Del Bimbo, *Imaging Novecento. A Mobile App for Automatic*

Recognition of Artworks and Transfer of Artistic Styles. Cham: Springer International Publishing, 2016, pp. 781–791. [Online]. Available: https://doi.org/10.1007/978-3-319-48496-9_62

[10] H. Becker, M. Naaman, and L. Gravano, "Beyond trending topics: Real-world event identification on twitter." *ICWSM*, vol. 11, no. 2011, pp. 438–441, 2011.

[11] M. Bertini, A. Del Bimbo, A. Ferracani, F. Gelli, D. Maddaluno, and D. Pezzatini, "Socially-aware video recommendation using users' profiles and crowdsourced annotations," in *Proc. of WSAM*, 2013. [Online]. Available: http://doi.acm.org/10.1145/2509916.2509924

[12] T. W. Bickmore, L. M. P. Vardoulakis, and D. Schulman, "Tinker: a relational agent museum guide," *Autonomous agents and multi-agent systems*, vol. 27, no. 2, pp. 254–276, 2013.

[13] A. D. Bimbo, M. Bertini, L. Seidenari, C. Baecchi, T. Uricchio, and A. Ferracani, "Portable computer vision for new "intelligent" audio guides," *EVA 2017 Florence - Electroning Imaging & the Visual Arts.*

[14] J. P. Bowen and S. Filippini-Fantoni, "Personalization and the web from a museum perspective," in *Proc. of Museums and the Web (MW)*, 2004.

[15] J. Brooke, "Sus-a quick and dirty usability scale," *Usability evaluation in industry*, pp. 189–194, 1996.

[16] ——, "Sus-a quick and dirty usability scale," *Usability evaluation in industry*, pp. 189–194, 1996.

[17] K. Bullington and J. Fraser, "Engineering aspects of tasi," *Transactions of the American Institute of Electrical Engineers, Part I: Communication and Electronics*, vol. 78, no. 3, pp. 256–260, 1959.

[18] J. Butterworth, A. Davidson, S. Hench, and M. T. Olano, "3dm: A three dimensional modeler using a head-mounted display," in *Proceedings of the 1992 Symposium on Interactive 3D Graphics*, ser. I3D '92. New York, NY, USA: ACM, 1992, pp. 135–138. [Online]. Available: http://doi.acm.org/10.1145/147156.147182

[19] G. Caggianese, L. Gallo, and P. Neroni, "Design and preliminary evaluation of free-hand travel techniques for wearable immersive virtual reality systems with egocentric sensing," in *Augmented and Virtual Reality.* Springer, 2015, pp. 399–408.

[20] A.-E. Cano, A. Varga, and F. Ciravegna, "Volatile classification of point of interests based on social activity streams." in *In Proceedings of the 10th International Semantic Web Conference, Workshop on Social Data on the Web (SDoW)*, 2011.

[21] M. Carter, F. Allison, J. Downs, and M. Gibbs, “Player identity dissonance and voice interaction in games,” in *Proceedings of the 2015 Annual Symposium on Computer-Human Interaction in Play.* ACM, 2015, pp. 265–269.

[22] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, “Return of the devil in the details: Delving deep into convolutional nets,” in *Proc. of BMVC*, 2014.

[23] C.-Y. Chen, B. R. Chang, and P.-S. Huang, “Multimedia augmented reality information system for museum guidance,” *Personal and ubiquitous computing*, vol. 18, no. 2, pp. 315–322, 2014.

[24] J. Chen, W. Geyer, C. Dugan, M. Muller, and I. Guy, “Make new friends, but keep the old: recommending people on social networking sites,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* ACM, 2009, pp. 201–210.

[25] Z. Cheng, J. Caverlee, K. Lee, and D. Z. Sui, “Exploring Millions of Footprints in Location Sharing Services,” in *Proceedings of the Fifth International Conference on Weblogs and Social Media.* Menlo Park, CA, USA: AAAI, Jul. 2011. [Online]. Available: http://www.aaai.org/ocs/index.php/ICWSM/ICWSM11/paper/view/2783

[26] G. Cirio, M. Marchal, T. Regia-Corte, and A. Lécuyer, “The magic barrier tape: A novel metaphor for infinite navigation in virtual worlds with a restricted walking workspace,” in *Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology*, ser. VRST ’09. New York, NY, USA: ACM, 2009, pp. 155–162. [Online]. Available: http://doi.acm.org/10.1145/1643928.1643965

[27] O. Cogal, V. Popovic, and Y. Leblebici, “Spherical panorama construction using multi sensor registration priors and its real-time hardware,” in *Multimedia (ISM), 2013 IEEE International Symposium on*, Dec 2013, pp. 171–178.

[28] E. Cohen, “The tourist guide: The origins, structure and dynamics of a role,” *Annals of Tourism Research*, vol. 12, no. 1, pp. 5–29, 1985.

[29] B. Craggs, M. Kilgallon Scott, and J. Alexander, “ThumbReels: Query sensitive web video previews based on temporal, crowdsourced, semantic tagging,” in *Proc. of CHI*, 2014. [Online]. Available: http://doi.acm.org/10.1145/2556288.2557249

[30] J. Cranshaw, R. Schwartz, J. I. Hong, and N. M. Sadeh, “The livehoods project: Utilizing social media to understand the dynamics of a city.” in *Proceedings of the Sixth International AAAI Conference on Weblogs an Social Media (ICWSM 2012)*, 2012.

[31] E. Crowley and A. Zisserman, “The state of the art: Object retrieval in paintings using discriminative regions.” in *BMVC*, 2014.

[32] P. Cui, Z. Wang, and Z. Su, “What videos are similar with you?: Learning a common attributed representation for video recommendation,” in *Proc. of ACM MM*, 2014. [Online]. Available: http://doi.acm.org/10.1145/2647868.2654946

[33] S. de Sousa Borges, V. H. Durelli, H. M. Reis, and S. Isotani, “A systematic mapping on gamification applied to education,” in *Proceedings of the 29th Annual ACM Symposium on Applied Computing*. ACM, 2014, pp. 216–222.

[34] T. Drugman, Y. Stylianou, Y. Kida, and M. Akamine, “Voice activity detection: Merging source and filter-based information,” *IEEE Signal Processing Letters*, vol. 23, no. 2, pp. 252–256, 2016.

[35] A. A. Efros and W. T. Freeman, “Image quilting for texture synthesis and transfer,” in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM, 2001, pp. 341–346.

[36] A. Elgammal and C.-S. Lee, “Separating style and content on a nonlinear manifold,” in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 1. IEEE, 2004, pp. I–478.

[37] V. Eliseo and L. Martine, “Ethnographie de l’exposition,” *Études et recherche, Centre Georges Pompidou, Bibliothèque publique d’information*, 1991.

[38] B. Elizalde and G. Friedland, “Lost in segmentation: Three approaches for speech/non-speech detection in consumer-produced videos,” in *Proc. of IEEE International Conference on Multimedia and Expo (ICME)*, 2013.

[39] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov, “Scalable object detection using deep neural networks,” in *Proc. of IEEE Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2014, pp. 2155–2162.

[40] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, “A density-based algorithm for discovering clusters in large spatial databases with noise.” in *Kdd*, vol. 96, no. 34, 1996, pp. 226–231.

[41] F. Eyben, F. Weninger, S. Squartini, and B. Schuller, “Real-life voice activity detection with lstm recurrent neural networks and an application to hollywood movies,” in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013.

[42] B. Fanini, E. d’Annibale, E. Demetrescu, D. Ferdani, and A. Pagano, “Engaging and shared gesture-based interaction for museums the case study of k2r international expo in rome,” in *2015 Digital Heritage*, vol. 1. IEEE, 2015, pp. 263–270.

[43] J. Feasel, M. C. Whitton, and J. D. Wendt, "Llcm-wip: Low-latency, continuous-motion walking-in-place," in *3D User Interfaces, 2008. 3DUI 2008. IEEE Symposium on*, March 2008, pp. 97–104.

[44] A. Felfernig, M. Jeran, G. Ninaus, F. Reinfrank, S. Reiterer, and M. Stettinger, "Basic approaches in recommendation systems," in *Recommendation Systems in Software Engineering*. Springer, 2014, pp. 15–37. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-45135-5_2

[45] A. Ferracani, L. Landucci, M. Faustino, G. X. Giannini, and A. Del Bimbo, "Natural experiences in museums through virtual reality and voice commands," in *Proceedings of the 25rd ACM international conference on Multimedia*. ACM, 2017.

[46] A. Ferracani, L. Landucci, and P. Pala, "Exploring 3d virtual environments through optimised spherical panorama navigation," in *IEEE International Conference on Multimedia and Expo (ICME 2015) - Demo Session*, no. USB Proceedings, 2015.

[47] A. Ferracani, D. Pezzatini, G. Becchi, L. Landucci, and A. Del Bimbo, "Separating the wheat from the chaff: Events detection in twitter data," in *Content-based Multimedia Indexing (CBMI 2017)*, 2017.

[48] A. Ferracani, D. Pezzatini, A. Benericetti, M. Guiducci, and A. Del Bimbo, "Pitagora: Recommending users and local experts in an airport social network," in *Proceedings of the 23rd ACM international conference on Multimedia*. ACM, 2015, pp. 755–756.

[49] A. Ferracani, D. Pezzatini, M. Bertini, and A. Del Bimbo, "Item-based video recommendation: An hybrid approach considering human factors," in *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval*. ACM, 2016, pp. 351–354.

[50] A. Ferracani, D. Pezzatini, M. Bertini, S. Meucci, and A. Del Bimbo, "A system for video recommendation using visual saliency, crowdsourced and automatic annotations," in *Proceedings of the 23rd ACM international conference on Multimedia*. ACM, 2015, pp. 757–758.

[51] A. Ferracani, D. Pezzatini, J. Bianchini, G. Biscini, and A. Del Bimbo, "Locomotion by natural gestures for immersive virtual environments," in *Proceedings of the 1st International Workshop on Multimedia Alternate Realities*, ser. AltMM '16. New York, NY, USA: ACM, 2016, pp. 21–24. [Online]. Available: http://doi.acm.org/10.1145/2983298.2983307

[52] ——, "Locomotion by natural gestures for immersive virtual environments," in *Proceedings of the 1st International Workshop on Multimedia Alternate Realities*. ACM, 2016, pp. 21–24.

[53] A. Ferracani, D. Pezzatini, and A. Del Bimbo, "User profiling for urban computing: Enriching social network trace data," in *Proceedings of the 3rd ACM Multimedia Workshop on Geotagging and Its Applications in Multimedia.* ACM, 2014, pp. 17–20.

[54] ——, "Roadie: Mobile semantic tourism routes," in *IEEE International Conference on Multimedia & Expo (ICME) - Demo Session*, no. USB Proceedings, 2015.

[55] A. Ferracani, D. Pezzatini, A. Del Bimbo, R. Del Chiaro, F. Yang, and M. Sanesi, "smart: Open and interactive indoor cultural data," in *Proceedings of the 23rd ACM International Conference on Multimedia*, ser. MM '15. New York, NY, USA: ACM, 2015, pp. 807–808. [Online]. Available: http://doi.acm.org/10.1145/2733373.2807981

[56] A. Fineschi and A. Pozzebon, "A 3d virtual tour of the santa maria della scala museum complex in siena, italy, based on the use of oculus rift hmd," in *3D Imaging (IC3D), 2015 International Conference on.* IEEE, 2015, pp. 1–5.

[57] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," *arXiv preprint arXiv:1508.06576*, 2015.

[58] D. Gavalas, V. Kasapakis, C. Konstantopoulos, K. Mastakas, and G. Pantziou, "A survey on mobile tourism recommender systems," in *Communications and Information Technology (ICCIT)*, 2013.

[59] T. Ge, L. Cui, B. Chang, Z. Sui, and M. Zhou, "Event detection with burst information networks." in *COLING*, 2016, pp. 3276–3286.

[60] R. Girshick, "Fast R-CNN," in *Proc. of IEEE International Conference on Computer Vision (ICCV)*, 2015.

[61] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. of IEEE Computer Vision and Pattern Recognition (CVPR)*, 2014.

[62] J. Gorham, "The relationship between verbal teacher immediacy behaviors and student learning," *Communication education*, vol. 37, no. 1, pp. 40–53, 1988.

[63] L. Han, A. Kashyap, T. Finin, J. Mayfield, and J. Weese, "Umbc ebiquity-core: Semantic textual similarity systems," in *Proceedings of the Second Joint Conference on Lexical and Computational Semantics*, 2013.

[64] Z. He, B. Cui, W. Zhou, and S. Yokoi, "A proposal of interaction system between visitor and collection in museum hall by ibeacon," in *Computer Science & Education (ICCSE), 2015 10th International Conference on.* IEEE, 2015, pp. 427–430.

[65] L. Hong, A. Ahmed, S. Gurumurthy, A. J. Smola, and K. Tsioutsiouliklis, "Discovering geographical topics in the twitter stream," in *Proceedings of the 21st International Conference on World Wide Web*, ser. WWW '12. New York, NY, USA: ACM, 2012, pp. 769–778. [Online]. Available: http://doi.acm.org/10.1145/2187836.2187940

[66] F.-H. Huang, "Motivations of Facebook users for responding to posts on a community page," in *Proc. of OCSC*, 2013. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-39371-6_4

[67] L. N. Huynh, R. K. Balan, and Y. Lee, "Deepsense: A gpu-based deep convolutional neural network framework on commodity mobile devices," in *Proc. of Workshop on Wearable Systems and Applications (WearSys)*, 2016.

[68] H. Iwata and Y. Yoshida, "Path reproduction tests using a torus treadmill," *Presence: Teleoperators and Virtual Environments*, vol. 8, no. 6, pp. 587–597, 1999.

[69] K. Järvelin and J. Kekäläinen, "Cumulated gain-based evaluation of ir techniques," *ACM Transactions on Information Systems (TOIS)*, vol. 20, no. 4, pp. 422–446, 2002.

[70] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM MM*. ACM, 2014, pp. 675–678.

[71] M. Jiang, P. Cui, R. Liu, Q. Yang, F. Wang, W. Zhu, and S. Yang, "Social contextual recommendation," in *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, ser. CIKM '12. New York, NY, USA: ACM, 2012, pp. 45–54. [Online]. Available: http://doi.acm.org/10.1145/2396761.2396771

[72] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," *arXiv preprint arXiv:1603.08155*, 2016.

[73] L. Johnson, S. Adams Becker, V. Estrada, and A. Freeman, *The NMC Horizon Report: 2015 Museum Edition.* ERIC, 2015.

[74] S. Karaman, A. D. Bagdanov, L. Landucci, G. D'Amico, A. Ferracani, D. Pezzatini, and A. Del Bimbo, "Personalized multimedia content delivery on an interactive table by passive observation of museum visitors," *Multimedia Tools and Applications*, vol. 75, no. 7, pp. 3787–3811, 2016.

[75] S. Karayev, M. Trentacoste, H. Han, A. Agarwala, T. Darrell, A. Hertzmann, and H. Winnemoeller, "Recognizing image style," *arXiv preprint arXiv:1311.3715*, 2013.

[76] J. Keil, L. Pujol, M. Roussou, T. Engelke, M. Schmitt, U. Bockholt, and S. Eleftheratou, “A digital look at physical museum exhibits: Designing personalized stories with handheld augmented reality in museums,” in *Proc. of Digital Heritage International Congress (DigitalHeritage)*, 2013.

[77] J.-S. Kim, D. Gračanin, K. Matković, and F. Quek, “Sensor-fusion walking-in-place interaction technique using mobile devices,” in *2012 IEEE Virtual Reality Workshops (VRW)*, March 2012, pp. 39–42.

[78] N. C. Krämer and G. Bente, “Personalizing e-learning. the social effects of pedagogical agents,” *Educational Psychology Review*, vol. 22, no. 1, pp. 71–87, 2010.

[79] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[80] T. Kuflik, Z. Boger, and M. Zancanaro, *Analysis and Prediction of Museum Visitors’ Behavioral Pattern Types.* Springer Berlin Heidelberg, 2012, pp. 161–176.

[81] J. E. Kyprianidis, J. Collomosse, T. Wang, and T. Isenberg, “State of the ”art”: A taxonomy of artistic stylization techniques for images and video,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 5, pp. 866–885, 2013.

[82] S. S. Latifi Oskouei, H. Golestani, M. Hashemi, and S. Ghiasi, “Cnndroid: Gpu-accelerated execution of trained deep convolutional neural networks on android,” in *Proc. of ACM Multimedia (MM)*, 2016.

[83] J. H. Lau, N. Collier, and T. Baldwin, “On-line trend analysis with topic models:\# twitter trends detection topic model online.” in *COLING*, 2012, pp. 1519–1534.

[84] K.-Y. Lin and H.-P. Lu, “Intention to continue using Facebook fan pages from the perspective of social capital theory,” *Cyberpsychology, Behavior, and Social Networking*, vol. 14, no. 10, pp. 565–570, 2011.

[85] D. A. Lindley, *This rough magic: The life of teaching.* JF Bergin & Garvey, 1993.

[86] M. Liu, X. Liu, Y. Li, X. Chen, A. G. Hauptmann, and S. Shan, “Exploiting feature hierarchies with convolutional neural networks for cultural event recognition,” in *Proc. of IEEE International Conference on Computer Vision (ICCV) Workshops*, 2015.

[87] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “Ssd: Single shot multibox detector,” in *Proc. of European Conference on Computer Vision (ECCV)*, 2016. [Online]. Available: http://arxiv.org/abs/1512.02325

[88] S. Livatino and C. Köffel, “Handbook for evaluation studies in virtual reality,” in *Virtual Environments, Human-Computer Interfaces and Measurement Systems, 2007. VECIMS 2007. IEEE Symposium on.* IEEE, 2007, pp. 1–6.

[89] X. Ma, H. Wang, H. Li, J. Liu, and H. Jiang, “Exploring sharing patterns for video recommendation on YouTube-like social media,” *Multimedia Systems*, 2013. [Online]. Available: http://dx.doi.org/10.1007/s00530-013-0309-1

[90] L. Malomo, F. Banterle, P. Pingi, F. Gabellone, and R. Scopigno, “Virtualtour: a system for exploring cultural heritage sites in an immersive way,” in *2015 Digital Heritage*, vol. 1. IEEE, 2015, pp. 309–312.

[91] P. Martin, B.-J. Ho, N. Grupen, S. Muñoz, and M. Srivastava, “An ibeacon primer for indoor localization: Demo abstract,” in *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, ser. BuildSys ’14. New York, NY, USA: ACM, 2014, pp. 190–191. [Online]. Available: http://doi.acm.org/10.1145/2674061.2675028

[92] V. F. Martins, P. N. Sampaio, F. d. S. Mendes, A. S. Lima, and M. de Paiva Guimarães, “Usability and functionality assessment of an oculus rift in immersive and interactive systems using voice commands,” in *International Conference on Virtual, Augmented and Mixed Reality.* Springer, 2016, pp. 222–232.

[93] K. Massoudi, M. Tsagkias, M. De Rijke, and W. Weerkamp, “Incorporating query expansion and quality indicators in searching microblog posts,” *Advances in information retrieval*, pp. 362–367, 2011.

[94] R. E. Mayer and R. Moreno, “Nine ways to reduce cognitive load in multimedia learning,” *Educational Psychologist*, vol. 38, no. 1, pp. 43–52, 2003.

[95] D. Metzler, C. Cai, and E. Hovy, “Structured event retrieval over microblog archives,” in *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.* Association for Computational Linguistics, 2012, pp. 646–655.

[96] A. Meyers, N. Johnston, V. Rathod, A. Korattikara, A. Gorban, N. Silberman, S. Guadarrama, G. Papandreou, J. Huang, and K. P. Murphy, “Im2calories: Towards an automated mobile vision food diary,” in *Proc. of IEEE International Conference on Computer Vision (ICCV)*, 2015.

[97] D. Milne and I. H. Witten, “Learning to link with Wikipedia,” in *Proc. of ACM CIKM*, 2008. [Online]. Available: http://doi.acm.org/10.1145/1458082.1458150

[98] A. Misra, “Speech/nonspeech segmentation in web videos,” in *Proc. of Annual Conference of the International Speech Communication Association (Interspeech)*, 2012.

[99] B. Mobasher, "Data mining for web personalization," *The adaptive web*, 2007.

[100] R. Morris, "Identity salience and identity importance in identity theory," *Current Research in Social Psychology*, vol. 21, no. 8, pp. 23–36, 2013.

[101] S. Mousazadeh and I. Cohen, "Ar-garch in presence of noise: parameter estimation and its application to voice activity detection," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 916–926, 2011.

[102] C. Munteanu, P. Irani, S. Oviatt, M. Aylett, G. Penn, S. Pan, N. Sharma, F. Rudzicz, R. Gomez, K. Nakamura *et al.*, "Designing speech and multimodal interactions for mobile, wearable, and pervasive applications," in *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems.* ACM, 2016, pp. 3612–3619.

[103] V. Navalpakkam and L. Itti, "Modeling the influence of task on attention," *Vision Research*, vol. 45, no. 2, 2005.

[104] F. Nayebi, J.-M. Desharnais, and A. Abran, "The state of the art of mobile application usability evaluation," in *Proc. of IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, 2012.

[105] S. a. A. Negrusa, Toader, "Exploring gamification techniques and applications for sustainable tourism," *Sustainability*, vol. 7, no. 8, p. 11160, 2015.

[106] D. T. Nguyen and J. E. Jung, "Real-time event detection for online behavioral analysis of big social data," *Future Generation Computer Systems*, vol. 66, pp. 137–145, 2017.

[107] J. Nielsen and R. Molich, "Heuristic evaluation of user interfaces," in *Proc. of SIGCHI Conference on Human Factors in Computing Systems (CHI)*, 1990. [Online]. Available: http://doi.acm.org/10.1145/97243.97281

[108] J. Norton, C. A. Wingrave, and J. J. LaViola Jr, "Exploring strategies and guidelines for developing full body video game interfaces," in *Proceedings of the Fifth International Conference on the Foundations of Digital Games.* ACM, 2010, pp. 155–162.

[109] A. Noulas, S. Scellato, C. Mascolo, and M. Pontil, "Exploiting semantic annotations for clustering geographic areas and users in location-based social networks," in *The Social Mobile Web*, 2011.

[110] A. Pagano, G. Armone, and E. De Sanctis, "Virtual museums and audience studies: the case of keys to rome exhibition," in *2015 Digital Heritage*, vol. 1. IEEE, 2015, pp. 373–376.

[111] K.-B. Park and J. Y. Lee, "Comparative study on the interface and interaction for manipulating 3d virtual objects in a virtual reality environment,"

Transactions of the Society of CAD/CAM Engineers, vol. 21, no. 1, pp. 20–30, 2016.

[112] S. Park, "The effects of social cue principles on cognitive load, situational interest, motivation, and achievement in pedagogical agent multimedia learning," *Journal of Educational Technology & Society*, no. 4, pp. 211–229, 2015.

[113] K.-C. Peng and T. Chen, "Cross-layer features in convolutional neural networks for generic classification tasks," in *Image Processing (ICIP), 2015 IEEE International Conference on.* IEEE, 2015, pp. 3057–3061.

[114] D. Picard, P.-H. Gosselin, and M.-C. Gaspard, "Challenges in content-based image indexing of cultural heritage collections," *IEEE Signal Processing Magazine*, vol. 32, no. 4, pp. 95–102, 2015.

[115] T. Piumsomboon, A. Clark, M. Billinghurst, and A. Cockburn, "User-defined gestures for augmented reality," in *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, ser. CHI EA '13. New York, NY, USA: ACM, 2013, pp. 955–960. [Online]. Available: http://doi.acm.org/10.1145/2468356.2468527

[116] A.-M. Popescu, M. Pennacchiotti, and D. Paranjpe, "Extracting events and event descriptions from twitter," in *Proceedings of the 20th international conference companion on World wide web.* ACM, 2011, pp. 105–106.

[117] F. Porikli, "Integral histogram: a fast way to extract histograms in cartesian spaces," *In Proceedings of IEEE CVPR, 2005*, 2005.

[118] Y. Qu and J. Zhang, "Trade area analysis using user generated mobile location data," in *Proceedings of the 22Nd International Conference on World Wide Web*, ser. WWW '13. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2013, pp. 1053–1064. [Online]. Available: http://dl.acm.org/citation.cfm?id=2488388.2488480

[119] S. B. Ranneries, M. E. Kalør, S. A. Nielsen, L. N. Dalgaard, L. D. Christensen, and N. Kanhabua, "Wisdom of the local crowd: Detecting local events using social media data," in *Proceedings of the 8th ACM Conference on Web Science.* ACM, 2016, pp. 352–354.

[120] S. Razzaque, D. Swapp, M. Slater, M. C. Whitton, and A. Steed, "Redirected walking in place," in *Proceedings of the Workshop on Virtual Environments 2002*, ser. EGVE '02. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2002, pp. 123–130. [Online]. Available: http://dl.acm.org/citation.cfm?id=509709.509729

[121] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. of IEEE Computer Vision and Pattern Recognition (CVPR)*, 2016.

[122] T. Reiners and L. C. Wood, Eds., *Gamification in Education and Business.* Springer International Publishing, 2015.

[123] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. of Advances in Neural Information Processing Systems (NIPS)*, 2015.

[124] Z. Rubin, "Disclosing oneself to a stranger: Reciprocity and its limits," *Journal of Experimental Social Psychology*, vol. 11, no. 3, pp. 233 – 260, 1975. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0022103175800254

[125] B. Ruf, E. Kokiopoulou, and M. Detyniecki, "Mobile museum guide based on fast sift recognition," in *International Workshop on Adaptive Multimedia Retrieval.* Springer, 2008, pp. 170–183.

[126] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes twitter users: real-time event detection by social sensors," in *Proceedings of the 19th international conference on World wide web.* ACM, 2010, pp. 851–860.

[127] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proc. of WWW*, 2001. [Online]. Available: http://doi.acm.org/10.1145/371920.372071

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

[129] ——, *Quantifying the user experience: Practical statistics for user research.* Morgan Kaufmann, 2012.

[130] B. Schilit, N. Adams, and R. Want, "Context-aware computing applications," in *Mobile Computing Systems and Applications, 1994. WMCSA 1994. First Workshop on.* IEEE, 1994, pp. 85–90.

[131] L. Seidenari, C. Baecchi, T. Uricchio, A. Ferracani, M. Bertini, and A. D. Bimbo, "Deep artwork detection and retrieval for automatic context-aware audio guides," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 13, no. 3s, pp. 35:1–35:21, Jun. 2017. [Online]. Available: http://doi.acm.org/10.1145/3092832

[132] P. Senin, "Dynamic time warping algorithm review," *Information and Computer Science Department University of Hawaii at Manoa Honolulu, USA*, vol. 855, pp. 1–23, 2008.

[133] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[134] K. M. Stanney and M. Zyda, "Virtual environments in the 21st century," *Handbook of virtual environments: Design, implementation, and applications*, pp. 1–14, 2002.

[135] F. Sulser, I. Giangreco, and H. Schuldt, “Crowd-based semantic event detection and video annotation for sports videos,” in *Proc. of CrowdMM*, 2014. [Online]. Available: http://doi.acm.org/10.1145/2660114.2660119

[136] E. A. Suma, D. M. Krum, and M. Bolas, *Human Walking in Virtual Environments: Perception, Technology, and Applications*. New York, NY: Springer New York, 2013, ch. Redirected Walking in Mixed Reality Training Applications, pp. 319–331. [Online]. Available: http://dx.doi.org/10.1007/978-1-4419-8432-6_14

[137] A. Sutcliffe and B. Gault, “Heuristic evaluation of virtual reality applications,” *Interacting with computers*, vol. 16, no. 4, pp. 831–849, 2004.

[138] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in *Proc. of IEEE Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9.

[139] R. Tanno, K. Okamoto, and K. Yanai, “Deepfoodcam: A dcnn-based real-time mobile food recognition system,” in *Proc. of International Workshop on Multimedia Assisted Dietary Management (MADiMa)*, 2016.

[140] F. Temmermans, B. Jansen, R. Deklerck, P. Schelkens, and J. Cornelis, “The mobile museum guide: artwork recognition with eigenpaintings and surf,” in *WIAMIS 2011, Delft, The Netherlands, April 13-15, 2011.* TU Delft; EWI; MM; PRB, 2011.

[141] J. B. Tenenbaum and W. T. Freeman, “Separating style and content with bilinear models,” *Neural computation*, vol. 12, no. 6, pp. 1247–1283, 2000.

[142] L. Terziman, M. Marchal, M. Emily, F. Multon, B. Arnaldi, and A. Lécuyer, “Shake-your-head: Revisiting walking-in-place for desktop virtual reality,” in *Proceedings of the 17th ACM Symposium on Virtual Reality Software and Technology.* ACM, 2010, pp. 27–34.

[143] W. M. Trochim *et al.*, “Likert scaling,” *Research methods knowledge base, 2nd edition*, 2006.

[144] L. Turchet, “Designing presence for real locomotion in immersive virtual environments: an affordance-based experiential approach,” *Virtual Reality*, vol. 19, no. 3-4, pp. 277–290, 2015.

[145] S. Unankard, X. Li, and M. A. Sharaf, “Emerging event detection in social networks with location sensitivity,” *World Wide Web*, vol. 18, no. 5, pp. 1393–1417, 2015.

[146] N. Vainstein, T. Kuflik, and J. Lanir, “Towards using mobile, head-worn displays in cultural heritage: User requirements and a research agenda,” in *Proceedings of the 21st International Conference on Intelligent User Interfaces.* ACM, 2016, pp. 327–331.

[147] J. Van Dijck, "'You have one identity': performing the self on Facebook and LinkedIn," *Media, Culture & Society*, vol. 35, no. 2, pp. 199–215, 2013.

[148] P. Vansteenwegen, W. Souffriau, G. V. Berghe, and D. V. Oudheusden, "The city trip planner: An expert system for tourists," *Expert Systems with Applications*, vol. 38, no. 6, pp. 6540 – 6546, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0957417410013230

[149] K. Vasylevska, H. Kaufmann, M. Bolas, and E. A. Suma, "Flexible spaces: Dynamic layout generation for infinite walking in virtual environments," in *3D User Interfaces (3DUI), 2013 IEEE Symposium on*, March 2013, pp. 39–42.

[150] F. Vesperini, P. Vecchiotti, E. Principi, S. Squartini, and F. Piazza, "Deep neural networks for multi-room voice activity detection: Advancements and comparative evaluation," in *Proc. of International Joint Conference on Neural Networks (IJCNN)*, 2016.

[151] M. Walther and M. Kaisser, "Geo-spatial event detection in the twitter stream." in *ECIR*. Springer, 2013, pp. 356–367.

[152] J. Wang, Y. Zhou, L. Li, B. Hu, and X. Hu, "Improving short text clustering performance with keyword expansion," in *The Sixth International Symposium on Neural Networks (ISNN 2009)*. Springer, 2009, pp. 291–298.

[153] Y. Wang, N. Stash, R. Sambeek, Y. Schuurmans, L. Aroyo, G. Schreiber, and P. Gorgels, "Cultivating personalized museum tours online and on-site," *Interdisciplinary Science Reviews*, vol. 34, no. 2-3, pp. 139–153, 2009.

[154] Z. Wang, J. Yu, Y. He, and T. Guan, "Affection arousal based highlight extraction for soccer video," *Multimedia Tools and Applications*, vol. 73, no. 1, 2014. [Online]. Available: http://dx.doi.org/10.1007/s11042-013-1619-1

[155] J. D. Wendt, M. C. Whitton, and F. P. Brooks, "Gud wip: Gait-understanding-driven walking-in-place," in *2010 IEEE Virtual Reality Conference (VR)*, March 2010, pp. 51–58.

[156] I. H. Witten and D. Milne, "An effective, low-cost measure of semantic relatedness obtained from wikipedia links," in *Proceeding of AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy, AAAI Press, Chicago, USA*, 2008, pp. 25–30.

[157] W. Woerndl and G. Groh, "Utilizing physical and social context to improve recommender systems," in *Proceedings of the 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Workshops*. IEEE Computer Society, 2007, pp. 123–128.

[158] K.-H. Woo, T.-Y. Yang, K.-J. Park, and C. Lee, "Robust voice activity detection algorithm for estimating noise spectrum," *Electronics Letters*, vol. 36, no. 2, pp. 180–181, 2000.

[159] B. Wu, E. Zhong, B. Tan, A. Horner, and Q. Yang, "Crowdsourced time-sync video tagging using temporal and personalized topic modeling," in *Proc. of ACM KDD*, 2014. [Online]. Available: http://doi.acm.org/10.1145/2623330.2623625

[160] J. Wu, C. Leng, Y. Wang, Q. Hu, and J. Cheng, "Quantized convolutional neural networks for mobile devices," in *Proc. of IEEE Computer Vision and Pattern Recognition (CVPR)*, 2016.

[161] C. Xia, R. Schwartz, K. Xie, A. Krebs, A. Langdon, J. Ting, and M. Naaman, "Citybeat: Real-time social media visualization of hyper-local city data," in *Proceedings of the 23rd International Conference on World Wide Web*. ACM, 2014, pp. 167–170.

[162] X. Xie, F. Tian, and H. S. Seah, "Feature guided texture synthesis (fgts) for artistic style transfer," in *Proceedings of the 2nd international conference on Digital interactive media in entertainment and arts*. ACM, 2007, pp. 44–49.

[163] P. Xu and M. Larson, "Users tagging visual moments: Timed tags in social video," in *Proc. of CrowdMM*, 2014. [Online]. Available: http://doi.acm.org/10.1145/2660114.2660124

[164] K. Yanai, R. Tanno, and K. Okamoto, "Efficient mobile implementation of a cnn-based object recognition system," in *Proc. of ACM Multimedia (MM)*, 2016.

[165] B. Yang, T. Mei, X.-S. Hua, L. Yang, S.-Q. Yang, and M. Li, "Online video recommendation based on multimodal fusion and relevance feedback," in *Proc. of ACM CIVR*, 2007.

[166] S. A. Yoon and J. Wang, "Making the invisible visible in science museums through augmented reality devices," *TechTrends*, vol. 58, no. 1, pp. 49–55, 2014.

[167] M. Zancanaro, T. Kuflik, Z. Boger, D. Goren-Bar, and D. Goldwasser, "Analyzing museum visitors' behavior patterns," in *Proc. of International Conference User Modeling (UM)*, 2007.

[168] Y. Zhang, S. Stellmach, A. Sellen, and A. Blake, "The costs and benefits of combining gaze and hand gestures for remote interaction," in *Human-Computer Interaction*. Springer, 2015, pp. 570–577.

[169] X. Zhao and S. E. Lindley, "Curation through use: Understanding the personal value of social media," in *Proc. of CHI*, 2014. [Online]. Available: http://doi.acm.org/10.1145/2556288.2557291

[170] Y. Zheng, L. Zhang, X. Xie, and W.-Y. Ma, “Mining correlation between locations using human location history,” in *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, ser. GIS ’09. New York, NY, USA: ACM, 2009, pp. 472–475. [Online]. Available: http://doi.acm.org/10.1145/1653771.1653847

[171] R. Zhou, S. Khemmarat, L. Gao, and H. Wang, “Boosting video popularity through recommendation systems,” in *Proc. of DBSocial*, 2011. [Online]. Available: http://doi.acm.org/10.1145/1996413.1996416

[172] C. L. Zitnick and P. Dollár, “Edge boxes: Locating object proposals from edges,” in *Proc. of European Conference on Computer Vision (ECCV)*, 2014, pp. 391–405.