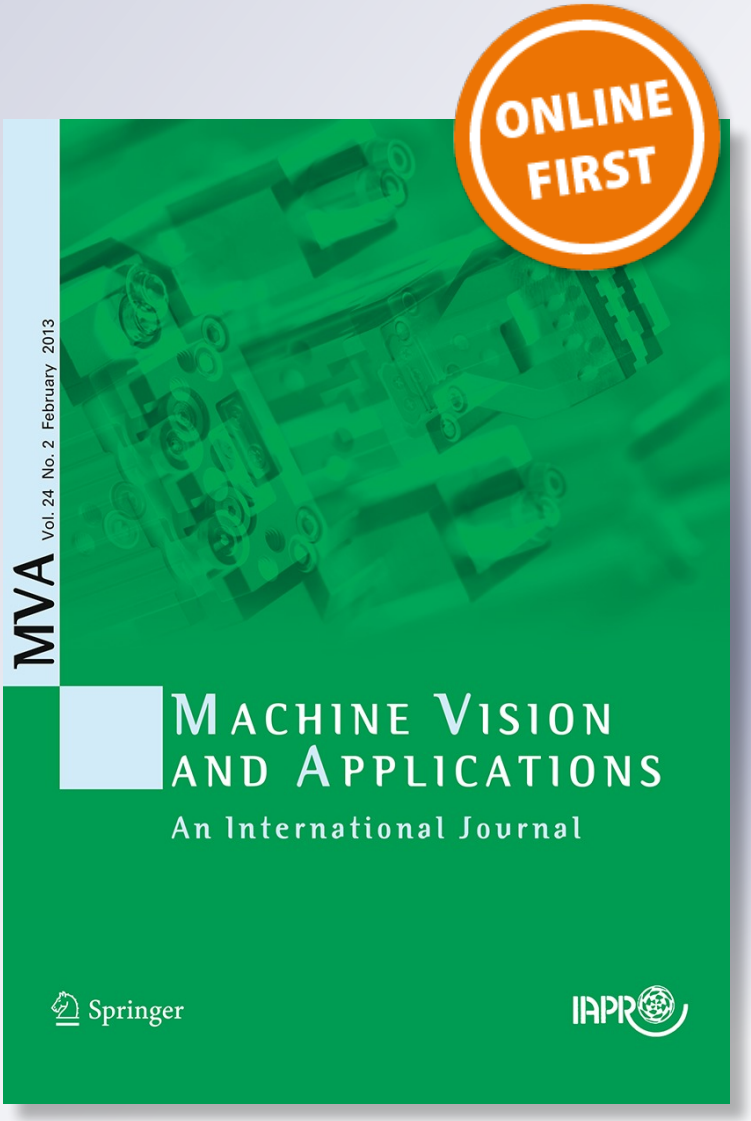
Using multiple sensors for reliable markerless identification through supervised learning

Andrea Albarelli, Filippo Bergamasco, Augusto Celentano, Luca Cosmo & Andrea Torsello

Machine Vision and Applications

ISSN 0932-8092

Machine Vision and Applications
DOI 10.1007/s00138-013-0492-2



Springer

Machine Vision and Applications
DOI 10.1007/s00138-013-0492-2

ORIGINAL PAPER

Using multiple sensors for reliable markerless identification through supervised learning

Andrea Albarelli · Filippo Bergamasco · Augusto Celentano · Luca Cosmo · Andrea Torsello

Received: 1 August 2012 / Revised: 19 December 2012 / Accepted: 13 February 2013

Abstract In many interaction models involving an active surface, there is a need to identify the specific object that performs an action. This is the case, for instance, when interactive contents are selected through differently shaped physical objects, or when a two-way communication is sought as the result of a touch event. When the technological facility is based on image processing, fiducial markers become the weapon of choice in order to associate a tracked object to its identity. Such approach, however, requires a clear and unoccluded view of the marker itself, which is not always the case. We came across this kind of hurdle during the design of a very large multi-touch interactive table. In fact, the thickness of the glass and the printed surface, which were required for our system, produced both blurring and occlusion at a level such that markers were completely unreadable. To overcome these limitations we propose an identification approach based on SVM that exploits the correlation between the optical features of the blob, as seen by the camera, and the data coming from active sensors available on the physical object that interacts with the table. This way, the recognition has been cast into a classification problem that can be solved through a standard machine learning framework. The resulting approach seems to be general enough to be applied in most of the problems where disambiguation can be achieved through the comparison of partial data coming from multiple simultaneous sensor readings. Finally, an extensive experimental section assesses the reliability of the identification.

Keywords Interactive surfaces · Human–machine interaction · Machine learning

1 Introduction

The Ca' Foscari University in the last years has issued several art exhibitions in its own premises, augmenting with multimedia technology the presentation of selected artworks for improving the visitors' satisfaction (Fig. 1).

Fig. 1 A group of people operating on the map-based multiuser art browser described in this paper

A. Albarelli (✉) · F. Bergamasco · A. Celentano · L. Cosmo · A. Torsello
Dipartimento di Scienze Ambientali, Informatica e Statistica, Università Ca' Foscari Venezia, Venice, Italy
e-mail: albarelli@unive.it

F. Bergamasco
e-mail: fbergama@dsi.unive.it

A. Celentano
e-mail: auce@dsi.unive.it

L. Cosmo
e-mail: lcosmo@dsi.unive.it

A. Torsello
e-mail: torsello@dais.unive.it

Published online: 13 March 2013

The early experiences [27] were built around a set of video installations distributed along the exhibition path, and interactive multimedia mobile devices to act as visitors' companions, providing contextual information to augment the visitors' knowledge about the exhibition contents. The overall initiative was part of a joint project, the *Interactive Multimedia Art Guide* project, involving the Department of Environmental Sciences, Informatics and Statistics, and the Department of Philosophy and Cultural Heritage. The outcomes of such a mixture of traditional museography and interactive multimedia have been evaluated, in terms of visitors' satisfaction, subjectively by questionnaires and objectively by tracing systems embedded in the mobile devices, logging user interaction styles and exploration of the exhibitions' content; evaluation summaries are reported in several papers describing the project [2,7,27].

Recently, the project took a new road. The exhibition "William Congdon in Venice (1948–1960): An American Look", held from May 5 to July 8, 2012, focused on the long stay of the artist, one of the protagonists of the American *Action Painting*, in Venice. The exhibition was showing the Congdon's paintings together with giant photos of selected artworks of his Venetian period, message boards presenting letters and sketches, and with the projection of drawings, notes and graphic works. The tight relationship between visual art and Venice, celebrated during several centuries by many artists, has led the curators to change the role of technology in the exhibition, moving from individual mobile information devices to a mix of private and shared experience in selected information areas: the *Venice Imago Project* aims at bridging the evolution of Venice in centuries with the artistic expressions, such as painting, photography and movies, depicting the town. To this end, three large interactive tables, each equipped with vision-based systems to track objects placed on it, have been built (Figs. 1, 2). The tables were decorated with maps of Venice pertaining to different historical and artistic periods: Sixteenth Century, Eighteenth Century, and the current days. The visitors could explore the city of Venice by placing and moving physical objects (cursors) on the map, activating the projection, on the wall surrounding the installation, of artworks, photographs and, for the more recent map, movies, related to the period referred by the map and to the location pointed by the cursor. On the Sixteenth Century table cursors were actually modified smartphones. These active devices were displaying information themselves and could also be used as personal devices for navigating the exhibition content independently from the table.

The project offered several challenges both on the museographic and on the technological side. From the museographic point of view, the presence of a shared interaction space required the definition of rules to associate the projections to the cursors' position and motion, allowing visitors to recognize the effect of their exploration. From the technological point of view, the size of the installation and the very dense and detailed maps decorating the tables posed serious constraints on the use of consolidated techniques based on fiducial markers; hence, a specially crafted technique was devised to associate the blobs as seen by the tracking system to the correct device.

In the following sections, after a review of the relevant literature, we present a detailed overview of the proposed setup. A first set of experiments will then point out the infeasibility of traditional marker-based recognition methods and thus, the need for an alternative identification technique. Subsequently, we give an in-depth description of the machine learning approach used to classify the observed blobs by virtue of the relations between positional information and sensor data. Finally, an extensive experimental section assesses the viability of the proposed multiple sensors approach and

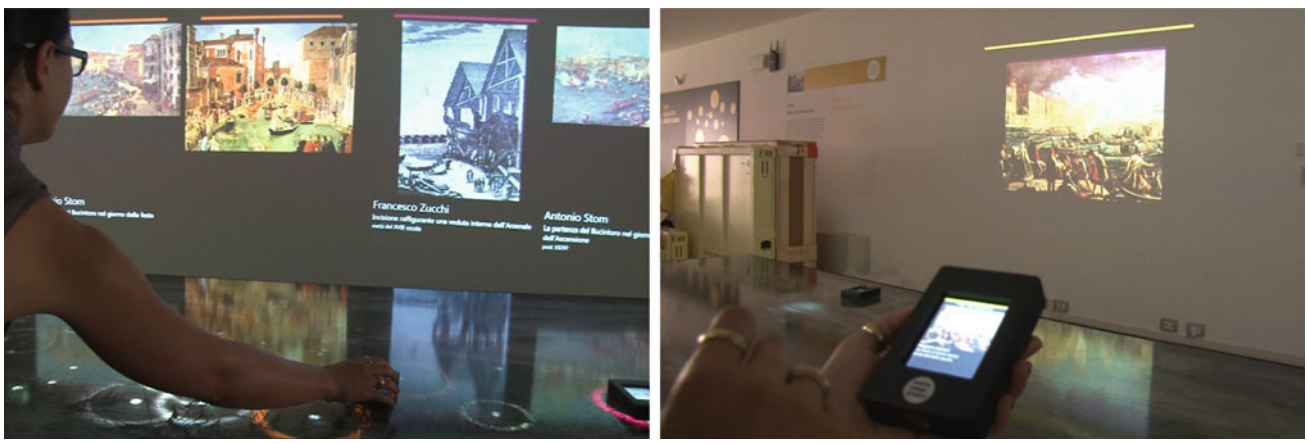


Fig. 2 Two close-ups of the table setup showing, respectively, the surface-based interaction mode (*left image*) and the device-based navigation (*right image*)

investigates some interesting aspects related to the synchronization between the different sources of data.

2 Related work

Interactive multiuser tables and walls have proved to be a viable system to foster user participation and interest in many shared environments, among which educational and cultural environments such as museums and exhibitions have a leading role. They favor interaction among users and induce a sort of serendipitous discovery of knowledge by observing the information exploration performed by other users. Their use has been analyzed and evaluated in entertainment as well as in educational applications [1,9,20,21].

Several technologies have been tested and evaluated in lab environments as well as in commercial products. Most of them only recognize and interpret the user touch and the placement of objects over the table, while the problem of associating the sensed information with specific users is solved in a few cases, often with sensors and equipment placed outside the table itself.

Among the early multitouch, multiuser surface implementations *DiamondTouch*™, an interactive table produced by Mitsubishi Electric Research Labs (MERL), was able to recognize up to four different users by matching signals captured at users' touch by small antennas placed under the table surface with receivers capacitively coupled to the users through their seats [12]. Such a user identification technique, while effective and robust, is oriented to a structured collaboration among the users and is not easily applied to highly dynamic environments like museums and exhibitions.

A different technology is used in the Microsoft *Surface*® interactive table, which uses five cameras and a rear projector to recognizes finger gestures as well as objects equipped with tags placed on the table surface. Gloves equipped with fiduciary tags are proposed by Marquardt et al. [25] to identify both what part of the hand and which user caused the touch.

The *frustrated total internal reflection* (FTIR) technology [17] has received greater attention as a cost-effective technology, able to trace many users with high frequency response; FTIR is based on infrared lateral illumination of a translucid surface, able to reveal small deformations caused by finger pressure. The problem of matching touches with users must, however, be solved by additional tracking systems based on analysis of the user position with additional external cameras and is subject to errors [14].

New opportunities for interaction in large shared spaces come from novel and promising methods for vision-based multitouch adopting depth sensors such as the broadly available Microsoft Kinect [13,24]. Initially dedicated to active gaming, Kinect is now often proposed as the key component of systems implementing a more "natural" interaction style. While arm and body gesturing is viable in many situations, it requires a clear identification of the user and his/her interaction space, and is therefore unsuitable in crowded spaces or when multiple users are involved. Moreover, external tracking devices must be placed over the table and might not be suitable for scenarios where a compact and self-contained system is needed. Further, even when specially crafted physical objects are used instead of hands or fingers, the depth-based tracking is harshly hindered by body and arm occlusion and does not support recognition out of the box.

In more recent years the widespread diffusion of mobile devices with rich interaction capabilities has suggested to couple personal (small) and public (large) screens for enhanced multiuser interaction. The personal devices are used both for input, allowing each of several users to provide direct interaction and own information to a shared system, and for output of local private information; the large shared screen acts as a collaboration and information sharing environment, guided by the input provided by the single users [15,16].

The Calisto system [5] is a multitouch kiosk which can share information with user personal *Android* devices; users can drag files and folders on the kiosk screen to *spotlets*, icons representing the personal devices connected; transfer occurs via HTTP and creates an information discovery environment shared among the kiosk users. Gestures on the personal devices are also interpreted to cause feedback from individual users to the shared kiosk. In this system, the identification of the user occurs when they connect to the kiosk. In [29] the coupling between a personal smartphone and a large shared screen occurs by touching icons with the smartphone and activating the flashlight. In [33] a shared environment is synthesized and projected on a wall by summing the actions performed by several users on their private devices. Users can select, upload and download objects in the shared space representing files, generating a collaborative environment. In such systems the user identification is easy because in the first two cases dedicated areas on the shared device are associated to individual private devices, while in the third system the shared space content is derived from the interaction only occurring on personal devices.

Since personal devices like smartphones and tablet computers are often equipped with a set of internal sensors, it has been quite natural for researchers to take advantage of them to expand the range of possible user interactions in new and very creative ways.

For instance, in [23] the author proposes to use the magnetic sensor to play virtual musical instruments with a touchless gesture-based interaction. This is possible as the user wears on his finger a little magnetic ring that exerts on the internal compass sensor an influence more significant than that of the Earth's magnetic field.

In [11], a technique that uses accelerometer data to build a gesture-based control schema is presented. Specifically tilt gestures are used to manage a continuous interaction between a mobile phone and a larger display, and throwing gestures are metaphors for transferring documents between a handheld device and a storage system.

The phone flashlight (when available) is used, besides in the already cited work by Schöning et al. [29], in [19,31] for light-based interaction between mobile phones and external large screens. In particular, the interaction can happen without the need for a wireless connectivity and thus is especially suitable for public spaces.

While in the aforementioned examples the sensors have been used with a direct mapping between the values gathered and the actions triggered, in many cases this relation is much less defined. This is especially true when complex gestures or patterns that are not entirely predictable must be recognized. In those scenarios, the preferred solution is often the use of machine learning techniques.

In [18], a large number of complex gestures is recognized by means of a fusion method based on features extracted from the time-domain and frequency-domain. Features are first fused together and dimensionally reduced through principal component analysis (PCA). Afterwards, a multi-class support vector machine (SVM) is used to classify the obtained vectors.

The authors of [35] address the problem of gesture recognition with a technique called frame-based descriptor and multi-class SVM (FDSVM). It is a *user-independent* approach that employs the SVM with a gesture descriptor combining spectral features and temporal features derived from 3D-accelerometers data. Gaussian random noise is added to the data to obtain a user-independent classification without the need of acquiring data from many different sources during the learning phase.

Finally, in [32], the sensor data are collected during an entire day of normal mobile phone usage. An SVM-based classifier is used to recognize many common physical activities with the aim of obtaining a complete monitoring of the user's lifestyle.

3 The context: a map-based multiuser art browser

The art exhibition mentioned in the introduction provided an opportunity to design and build three large (3×2 mt) interactive tables equipped with the standard set of input and output devices such as cameras for blob detection and projectors for information display. While a focused effort has been made in order to create a generic and reusable system, the design of the table hardware and software is still the result of requirements partly bound to the interaction functions, partly imposed by the environment.

3.1 Interaction model

Each table presents a high resolution diaphanous map of Venice (Fig. 3-2) printed on a thick glass surface (Fig. 3-1). A total of three tables has been built. Each one portraits a different period of the city history. The well-known lithography made by Jacopo De' Barbari [28] that represents an aerial view of the island of Venice has been selected to represent the Sixteenth Century. The Napoleonic cadastral map was chosen to provide an overview of the city during the Eighteenth Century. Finally, a satellite view has been used to represent the modern era.

The required interaction is based on placing or moving on the table objects representing the virtual visitor position in the town. These objects, that in the following will be referred to as *cursors*, are smartphones that are equipped both with a display and some internal sensors, such as accelerometers and compass (Fig. 3-3).

Relevant places are associated to paintings by artists of different epoques, portraying the city views related to the location selected by the user and the historical period expressed by the specific map. They are shown as pulsating spots on the map, attracting the user attention (Fig. 4a).

As the user moves the cursor over a relevant place, the spot is highlighted to confirm its detection and the corresponding artwork is projected on the walls surrounding the installation (Fig. 3-5); relevant information about the author and the picture are presented on the display of the cursor.

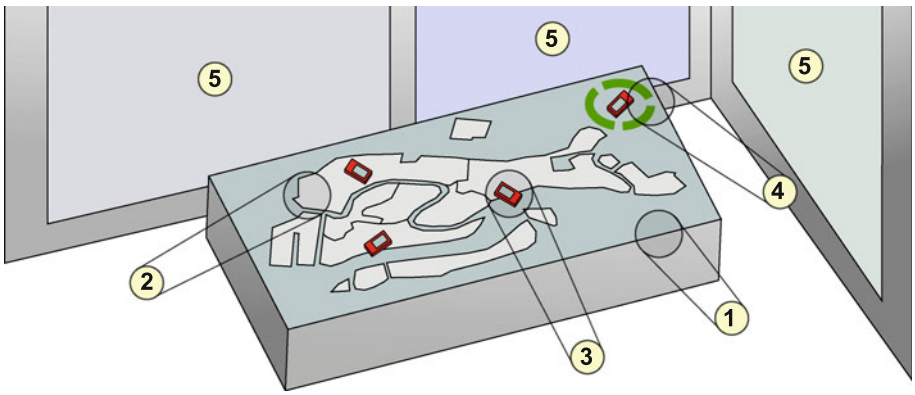


Fig. 3 Schematic representation of the components of the proposed multiuser interactive table

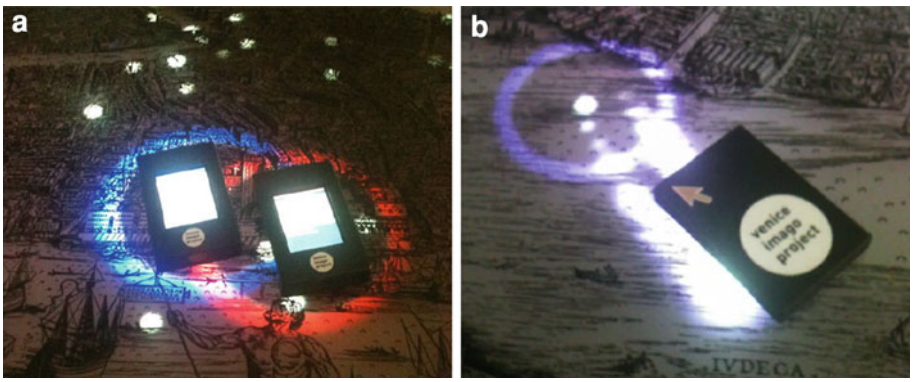


Fig. 4 **a** The feedback on cursor location. **b** A visual suggestion to move

Further, the cursors can be lifted from the table and used as a gesture-based remote control that allows the user to get near the projected paintings and still continue the browsing activity, updating the projection.

To allow more users to interact with the table and to experiment a simpler interaction style, suitable for less skilled users, a few *passive* cursors have also been used: they are small rectangular boxes decorated with the project logo, which can be placed and moved on the table like the active cursors, causing the same behavior of the active devices. In this case, obviously, there is no information processing and display on the cursor.

Additional visual cues are generated if the cursor is placed near to a spot but not directly over it. A stream of light is generated, moving from the cursor to the nearest spot, suggesting a move (Fig. 4b).

Each user is independent; at most 5–6 users can operate at the same time, distributed along the table sides with a comfortable amount of surrounding space to experience an open view of a part of the map. Such distribution assures also that the user position around the table allows the placement of the video projections on the walls in a regular and pleasant layout.

Such layout is automatically arranged by an algorithm that tries to optimize space usage by dynamically resizing pictures and trying to place a new artwork directly in front of the user that required it. Since several new paintings could appear at the same time, special care must be applied to ensure that the user receives enough cues to visually associate the newly displayed painting with the action he/she just completed.

The tables are operated by the users without help, but the installations are guarded by cultural mediators, personnel available to visitors to help them in case of need, and ready to explain both the table functions and the associated content.

3.2 Optical system

As for any multitouch system, one of the most critical choices is related to the technology used to detect the user interaction with the table surface. Given the large active surface (measuring 300 × 200 cm) using a touch sensitive overlaid plane was not an option for economic and practical reasons. Also the adoption of an external vision-based tracking system was not viable, since the large number of simultaneous users would cause unpredictable occlusion conditions.

In this regard, our choice has been directed to classical blob detection by placing a number of cameras inside the table and oriented toward the surface. Specifically, we used infrared cameras (i.e. cameras that leave out the visible light) equipped with an 850 nm thresholded filter.

This kind of camera is usually coupled with some source of infrared illumination, so that the visible light produced by the internal projector does not interfere with the blob-detection. To this end, two illumination techniques are usually adopted: the frustrated total internal reflection (FTIR) and the diffused illumination (DI).

FTIR is based on the interference between the object in contact with the surface and infrared light tangentially diffused inside the thickness of the surface layer and trapped by the total reflection angle. Such interference causes the light to change direction and thus to escape from the surface layer toward the camera. By contrast DI implies the naïve illumination of the objects via a diffused light that passes through the surface and is reflected as it encounters an infrared-reflective obstacle.

For this installation we used DI, which performed better than FTIR, as the large size of the table hampers the even diffusion of the light and the strength of the returned signal. This limitation is even more exacerbated by the presence on the lower side of the table surface of a diffuser layer that is needed to offer a screen for back-projection. The visualization itself happens by means of two short throw projectors mounted inside the table.

Given the unfavourable ratio between the size of the table and its height (about 85 cm), the use of a first-surface mirror has been necessary to create a suitable light path. The cursors consist of six Android-based phones that communicate with the PC controlling the business logic via Bluetooth.

4 Contribution of the work

This paper gathers two different contributions which, despite being tightly coupled within the overall problem, are very different in nature.

The first topic is related to the above-described multiuser art browser based on a map metaphor. The designed interaction model provides to the user the ability of selecting different artworks by placing and moving an active physical device on the surface of a map. Different users can select different artworks, which are displayed around the table. The device can also be lifted from the table and used to continue the interaction in a private way, using conventional touch gestures on the device display to move between artworks or to gather further information about a specific painting.

For this model to be correctly implemented the application logic must be continuously aware of the connection between the touch events and the cursor that generated such events. In fact, regardless of the technology used to detect touches and to communicate with the cursors, each time a user places his/her cursor on the table, the associated device must be notified about the selected coordinates in order to be able to update its status.

Within our specific setup the spatial position of the cursor is detected by a camera placed inside the table, thus the problem reduces to associate each blob seen to a device. A straightforward solution to this problem would be to place different fiducial markers on the bottom side of different devices. Unfortunately, as thoroughly investigated in the following (see Sect. 5), this is not feasible in this installation, as the thickness of the glass and the presence of a semi-opaque projection surface make impossible to distinguish anything more significant than the blurred contours of the cursor.

Since each device is equipped with accelerometers, it seems to be reasonable to use them to relate acceleration data coming from the cursor to the movements observed by the camera. In principle this could be done easily by instructing the user to perform some specific movement pattern during initialization, but this approach would suffer from several limitations: it is an intrusive technique and relies on the ability and willingness of the user to perform an initial calibration, which is not feasible in an art exhibition context with casual users. Further, ambiguity could still arise due to gesture errors or to simultaneous initialization by different users. Finally, the connection between blobs and devices would hold only as long as the cursors are perfectly tracked by the camera and would break as the device is lifted by the user, thus requiring a new initialization.

To solve these shortcomings we resorted to a different approach, which represents the second, and more technically oriented, contribution of this paper. Namely, we adopted a SVM classifier to tell if a blob is related to a stream of accelerometer data or not. The classifier is initially trained with a large set of both positive and negative examples and it is then used to classify in real time each blob seen by the camera with respect to the recent sensors history of all the active devices. This way the association is fully automated, continuously refined and does not require any action by the user.

While this solution could seem to be specially crafted to solve the specific technical problem that arises from the interaction model and the installation context, we do really feel that it is general enough to be applied to many different scenarios. In fact, the ability of labelling optically undistinguishable objects by virtue of sensor data can be useful in any situation where the line of sight is hindered and dead reckoning is not accurate enough over long stretches of time. These settings include, for instance, the recognition of soccer players during a game, the labelling of forklifts in a warehouse or even the tracking of shopping carts in a mart for marketing analysis.

5 Unreliability of marker recognition

Since the proposed interaction model requires to associate each blob seen by the camera to a device, a reliable identification schema must be deployed. The adoption of diffused illumination would normally allow the use of fiducial markers to perform recognition. For this reason our first prototype was based on ARToolkit+ [34] (see Fig. 6), a widely used extension to the well-known ARToolkit tag system [22].

Unfortunately, in our setup we faced several hurdles that compromised the viability of a marker-based recognition. The first problem is related to the size of the table. In fact, for the surface to be sturdy enough to be safe and do not flex under its weight, it has been necessary to use a glass pane 12 mm thick. Since the diffuser layer is placed on the bottom side of the surface, this resulted in the marker printed surface being at least 12 mm away from it, which in turn caused a strong blurring. This blur does not prevent to track the objects as blobs, however, it inhibits even the most infrared-reflective markers to be distinguished reliably. Further, an

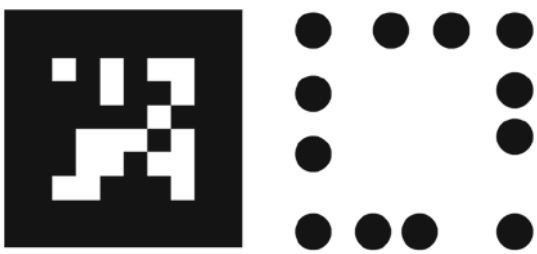


Fig. 5 The two fiducial marker designs tested with our setup: ARToolkit+ (*on the left*) and Pi-Tag (*on the right*)

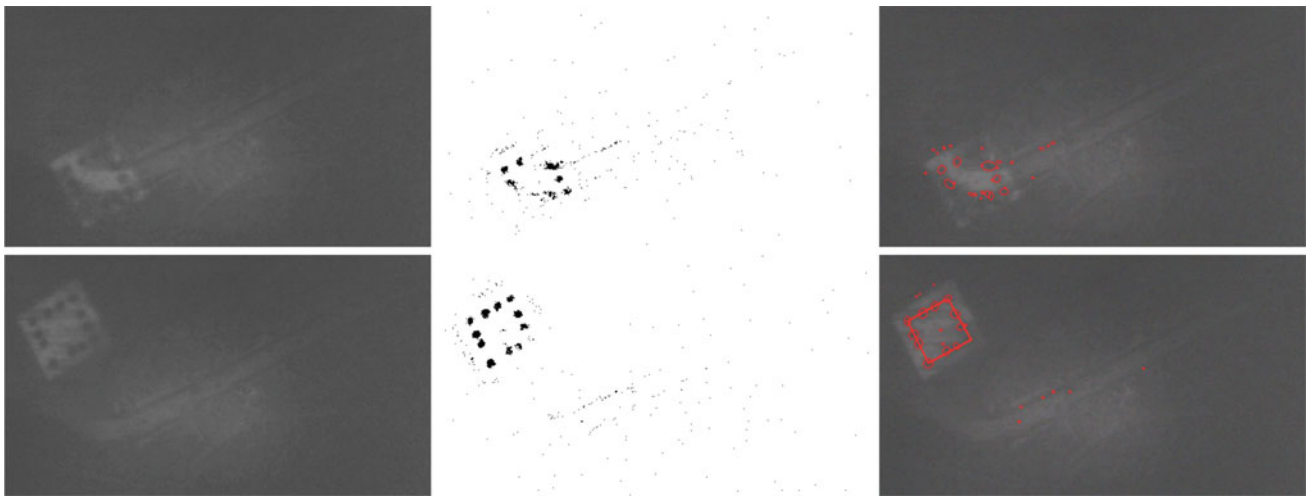


Fig. 6 Two examples of the Pi-Tag recognition process with our setup. The original shot is shown in the *first column*. The *other two columns* show, respectively, the thresholded image and the detected ellipses (bad contrast is due to the low transmittance of the glass)

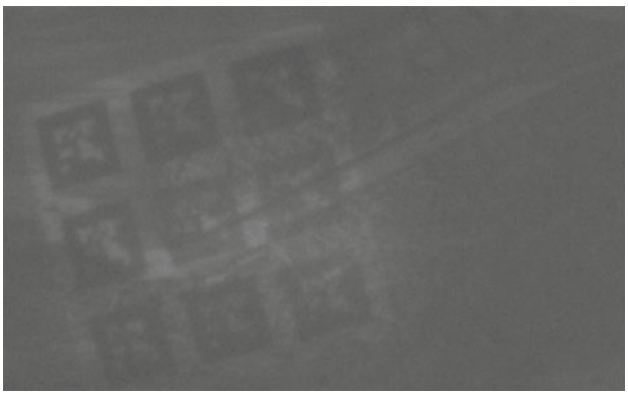


Fig. 7 The effects of glass thickness and surface print over the readability of ARTookit+ tags (bad contrast is due to the low transmittance of the glass)

additional source of noise and occlusion is represented by the map printed on the upper side of the surface.

In order to minimize such negative effects, we took several measures: the markers have been printed on a substrate highly reflective with respect to infrared light, the light spots have been carefully calibrated to avoid blooming and reflections while still allowing an even and bright illumination, the camera exposure and gain have been manually optimized to get the best compromise between signal and noise. Finally we adopted a best-of-breed adaptive thresholding algorithm and we manually tuned it to get the best foreground/background separation (Fig. 5).

In spite of these precautions, ARToolkit+ was not able to correctly recognize its markers, which appeared very faint and occluded to the camera (see Fig. 7).

As a last resort, we tried to change the fiducial marker system and we made some in-depth experiments using Pi-Tag, a recently introduced fiducial tag that exhibits an ellipse-based design that is moderately resilient to occlusion [4]. The adoption of an ellipse-based design makes a lot of sense within our setup, since ellipse detection is fairly robust to isotropic noise such as blur. Further, the Pi-Tag recognition algorithm is able to cope with some missing ellipses, which could help a lot when dealing with the non-uniform occlusion caused by the printed map overlay.

Regarding this latter problem, our first batch of qualitative tests revealed that the recognizability of the markers strongly depends on the local density of the printed overlay. In Fig. 6 we show two anecdotal examples. In the first one, the marker is seen through a non-uniform area where a Venice "canal" separates two blocks of buildings (see also Fig. 8 for a bitmap image of the printed area). In this case only the ellipses on the clear area can be detected and the combination of blur and occlusion is too strong to allow recognition. By contrast, in the second example shown in Fig. 6, the marker is placed on the clear area of the "lagoon" and, while an ellipse is still missing, the remaining signal is good enough for the tag to be detected and recognized.

Even from this simple qualitative evaluation it seems that the quality of recognition is still too unpredictable to be deemed as reliable. To get a quantitative assessment of this speculation we made a video a couple of minutes long that captures the marker moving over several locations. To obtain a fair evaluation for our setup, we tried to evenly cover the area. In Fig. 8 we plotted the location of the detected markers over a part of the original bitmap of the printed map. As expected, it can be observed that most of the successful recognitions happen within the less occluded areas. This behaviour is further characterized in the second column of Fig. 8. In this histogram recognition events are grouped according to the average grey level exhibited by the map area where they happen. Finally, the over-

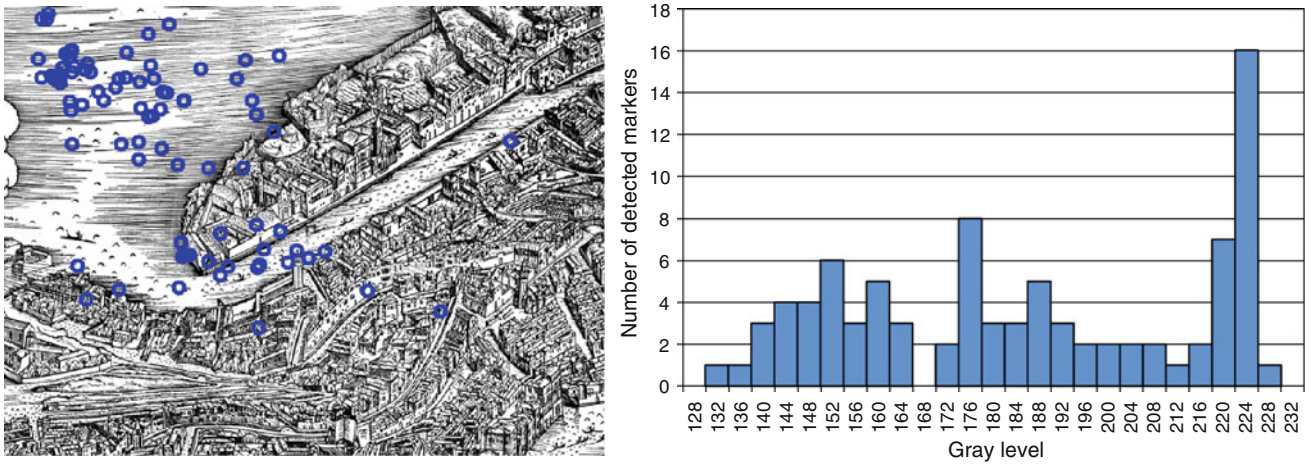


Fig. 8 The locations of the detected markers in the test video overlayed to the printed map (*on the left*) and the distribution of the detections with respect to the average gray level

all recognition rate was about 1 %, since we obtained only 89 correct detections within a video with more than 7,000 frames.

This very low level of reliability of the marker-based recognition, prompted us to explore different approaches in order to associate a blob to the device that produced it. Specifically, we exploited the expected correlation between blob movements (as seen from the camera) and acceleration data (as measured by the device sensors). The details of the implemented solution and its effectiveness in a real scenario are described in the following sections.

6 Cursor/table communication

Before addressing the task of associating each tracked cursor to its identity, we first need to sort out a couple of preliminary problems related to the data interchange between the active cursors and the table itself. Namely, we have to define a protocol for message transmission and a technique to obtain a reliable synchronization between the real time clock of the cursor and of the table. The latter is especially important, since we will adopt machine learning techniques that will relate specific data patterns gathered from the sensors with the information obtained from the camera. If proper synchronization does not happen, both the learning and the recognition steps can be severely hindered since the mutual causality between the two phenomena could not hold any more.

6.1 Message exchange

All the communications throughout the system happen by exploiting the serial port profile (SPP) of the Bluetooth standard. From a design point of view this is a reasonable choice for many reasons. For starters, Bluetooth requires much less power than Wi-Fi to work, and since the cursors should be able to run on battery power for a whole 8-h day, energy saving must be seriously taken in account. Specifically, the device creates the server SPP socket, i.e. it presents itself as a serial port service in a similar way to what external GPS antennas or barcode readers generally do. Each device is first paired with the table, which scans at intervals for them. When a device is found, the table initiates the serial connection. The lack of connection for a long period indicates that a device is either malfunctioning or has been stolen, either way, a warning should be triggered. The communication protocol uses Consistent Overhead Byte Stuffing [8] to transmit packets made up of a header, that specifies a message type and the id of the sender, and a payload that is defined according to the characteristic of the exchanged data. There is a total of five types of messages that are transmitted within the system:

Type	Sender	Content
Sensors	Device	Accelerometer and compass data
Url	Table	Url of the content to display
Action	Device	Url selected by the user
Ping	Table	Timestamp of table real time clock
Pong	Device	Timestamp of device real time clock

The *Sensors* message is sent at regular intervals from the device to the table and contains the data gathered from the accelerometers and the magnetic field sensor. Since the actual update frequency of such sensors is usually very high on most Android devices, the data are not sent directly, rather an integration step is performed on board to make the update rate of the sensor data commensurate to the frame rate of the infrared camera (about 30 fps). The integration step implies the additional advantage of an implicit noise reduction due to the averaging.

The *Url* message is sent by the table to control the display of the device. Each device contains a set of HTML pages that can be loaded by the local browser. Once the table identifies a device, it sends the Url that selects a menu page related to the area of the map where the device has been placed. This is the only control action that the table performs with respect to the device.

The *Action* message is sent from the device when a user clicks on a link in the local browser. The click is intercepted by the application running on the device and the GET parameters (if any) are sent to the table. This protocol allows to define custom parameters to trigger actions by the table such as the display of an artwork, the highlight of interesting point on the map or any other interaction that can be added in the future.

Finally, the *Ping* and *Pong* messages are used to transmit the current time (in milliseconds) as measured by the real time clock, respectively, of the table and of the device. These two message are meant to be used to perform a round trip, initiated by the table, for internal clock synchronization. The details about how this synchronization happens will be given in the next section.

6.2 Time synchronization

In order to properly correlate the data coming from all the devices we must be able to measure the value of all the sensors at certain times. Even if the delay from the camera acquisition to the blob identification is less than a couple of milliseconds and can be ignored, this is not true for the data coming from the devices accelerometers. Indeed, the delay introduced by the Bluetooth communication is in the order of tens of milliseconds and also, unfortunately, is not constant, so we cannot reliably compute the data time from the arrival time measured at the table PC. The importance of automatically synchronize the clock of all devices is twofold. First, the initial offset between each clock is not negligible and must be taken into account to properly estimate the status of the system. Second, due to the low accuracy of the quartz clock on modern devices that do not tick exactly with the same frequency, a continuously increasing drift is accumulated as time goes by.

To estimate with high accuracy the time offset between the server and each device, we chose to adopt the same method used in SNTP. The synchronization process starts by collecting a set of pairs (o, d) where o is the offset between the server and a specific device, and d is the measured round trip time. To gather each of this data, a packet is sent to the device containing the current value of server time. On arrival, the device must reply attaching to the packet its own time. When the server receives the reply, is able to compute the offset between its time with respect to the device and the round trip time as the difference between the arrival time and the original send time contained into the packet. If we assume a symmetric unknown communication delay, the computed offset would be exact, but this assumption is just unfeasible for Bluetooth communication devices. However, we can state for sure that the true offset value must be contained in the interval $[o - d/2...o + d/2]$. To restrict the interval as much as possible, a lot of (o, d) pairs are collected and the largest common interval (see Fig. 9) is computed by using the algorithm proposed in [26].

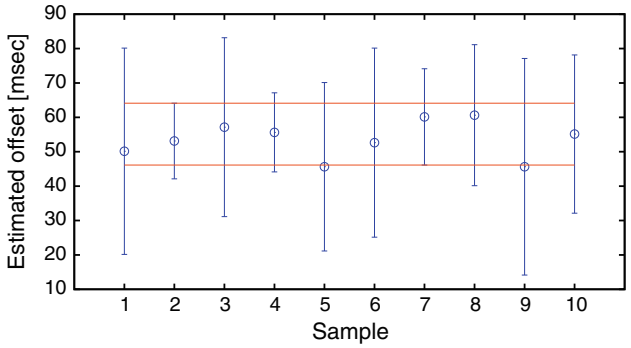


Fig. 9 Example of the offset estimation via intersection of measured intervals

7 Identification by learning

In principle, several techniques could be adopted to identify blobs by combining sensor data and tracking information. For instance, a hand crafted decision tree with suitable thresholds could be applied to perform a direct verification of the compatibility between the blob status and the reported orientation of the compass. However, this kind of approach becomes cumbersome when the amount of information begins to grow, which is the case, for instance, if the history spanning the last few frames is considered. Further, it is not always obvious how to relate the data and how to weight the contribution of each source of information. Whenever a con-causal relation between different data sources exists, but it is not clear how to design and parametrize a direct algorithm to exploit such relation, resorting to some machine learning technique is a natural choice. In fact, given a reasonable feature selection, learning techniques have proven to be often more effective in classification tasks than manually crafted solution that exploit a direct knowledge of the problem domain [3,30].

We decided to address the issue of blob-device association in terms of a non-probabilistic binary classification problem. In fact, during the normal usage of the system, two crucial class of events can occur in which inference from sensor data can be performed to disambiguate some of the pairs. In the following we will refer to these two events with the terms *appear* and *stop*. The *appear* event happens when a new blob starts to be tracked by the system. If the blob is generated by one of the connected devices, the time-synchronized

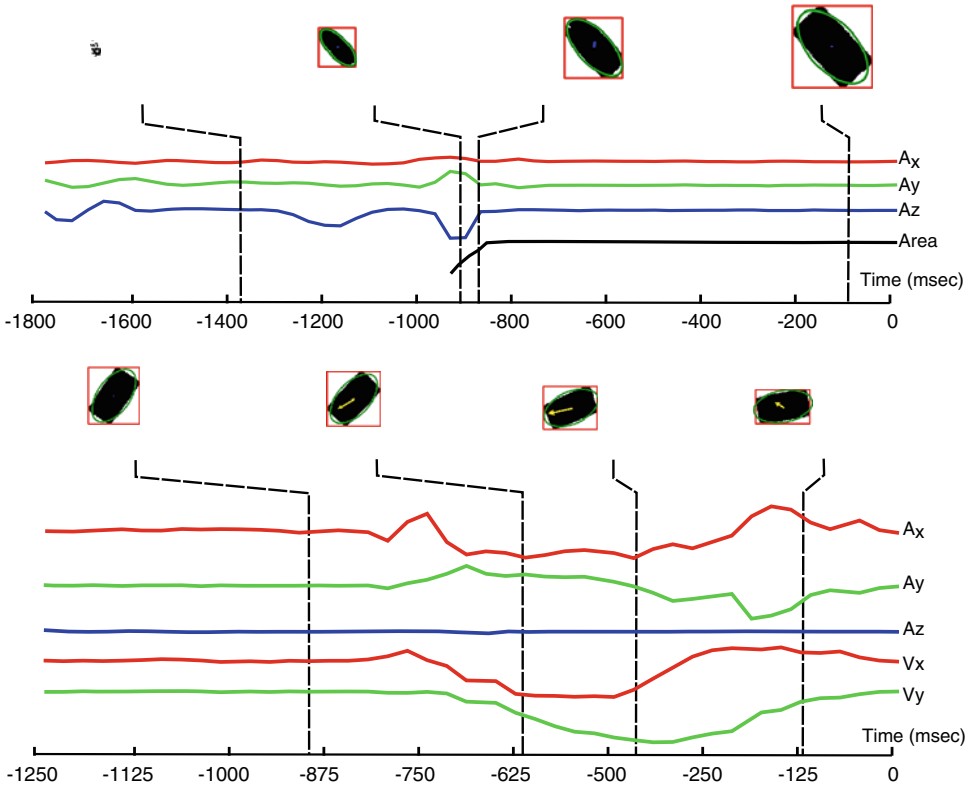


Fig. 10 Mutual causality relations between the observed blobs and the data gathered from the sensors. In the *first row* an *appear* event is shown: note that as the object touches the table a sudden stop in the vertical acceleration (Az) can be observed, as well as a fast increase in the blob area. In the *second row* a stop event is detected as the accelerations (Ax, Ay, Az) and the velocities (Vx, Vy) measured toggle from a quiet state to an active state and then to a quiet state again. Note that the speed of the blob measured from the camera (the *arrow* in the blob) agrees with the data coming from the sensors

signal produced by the accelerometers should be somehow related to the increasing area of the blob. Also, the absolute orientation of the device with respect to the magnetic north should correspond to a specific orientation of the blob in the image frame (assuming that the table is not moved once calibrated). Differently, the event *stop* is triggered when a tracked blob stops moving. When this occurs, the blob velocity signal computed by the table will probably be coherent with the accelerometer data, both defining the same space-time trajectory.

In Fig. 10 an example of the signals coming from the aforementioned sensors is shown for an instance of the *appear* and *stop* events. Because of the non negligible sensor data correlation that is exhibited in this two particular events, a binary classifier should be able to answer the question "Is this blob data related to this specific device data?" with very high accuracy. Many different types of binary classifiers have been proposed in literature, each with its own strengths and weakness. Due to the relatively high-dimensional well-separable sensor data we decided to use the well-known *support vector machine* method [6].

7.1 Machine learning with SVM

Suppose that we are given training data

$$\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_\ell, y_\ell)\} \subseteq \mathbb{R}^d \times \{-1, 1\} \tag{1}$$

where $\mathbb{R}^d$ denotes the space of the input patterns (sensor data in our case) and $y_i \in \{-1, +1\}$ indicates the binary class to which the point $\mathbf{x}_i$ belongs. In the simplest case, we can assume that there exist some hyperplanes which separate all points having $y_i = 1$ (positive examples) from those having $y_i = -1$ (negative examples). Any of those hyperplanes can be defined as the locus of points $\mathbf{x}$ satisfying the equation.

$$\mathbf{w} \cdot \mathbf{x} + b = 0 \tag{2}$$

where $\mathbf{w}$ is the normal of the hyperplane and $|b|/\sqrt{\mathbf{w} \cdot \mathbf{w}}$ is the perpendicular distance from the hyperplane to the origin. A support vector algorithm simply looks for the hyperplane that maximizes the margin with respect to all points, defined as the shorted distance between the hyperplane and any of the negative or positive point. If the training data are linearly separable (this hyperplane exists), one can find a pair of hyperplanes such that no point lies between them and the following constraints are satisfied:

$$\mathbf{x}_i \cdot \mathbf{w} + b \geqslant +1 \qquad \forall y_i = +1 \tag{3}$$

$$\mathbf{x}_i \cdot \mathbf{w} + b \leqslant -1 \qquad \forall y_i = -1 \tag{4}$$

that can be combined into:

$$y_i (\mathbf{x}_i \cdot \mathbf{w} + b) - 1 \geqslant 0 \qquad \forall i \tag{5}$$

It is easy to demonstrate that the hyperplane with largest margin can be found by solving the following convex optimization problem:

$$\begin{aligned} &\text{minimize } \tfrac{1}{2}||\mathbf{w}||^2 \\ &\text{subject to } y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 \geqslant 0 \quad \forall i \end{aligned} \tag{6}$$

This problem is feasible only under the assumption that such hyperplane actually exists. However, it may not be the case even if we know that the data should be linearly separable considering the presence of outliers or noise that may hinder that assumption.

To this extent it is common to relax the constraints (3) and (4) by introducing positive slack variables $\xi_i, i = 1, \ldots, \ell$ transforming the formulation with the one proposed in [10]:

$$\begin{aligned} &\text{minimize } \tfrac{1}{2}||\mathbf{w}||^2 + C\sum_{i=1}^{\ell}\xi_i \\ &\text{subject to } \mathbf{x}_i \cdot \mathbf{w} + b \geqslant +1 - \xi_i \quad \forall y_i = +1 \\ &\qquad\qquad \mathbf{x}_i \cdot \mathbf{w} + b \leqslant -1 + \xi_i \quad \forall y_i = -1 \\ &\qquad\qquad \xi_i \geqslant 0 \quad \forall i \end{aligned} \tag{7}$$

Roughly speaking, the constant $C > 0$ is a weighting term that determine how much we are interested to keep the hyperplane flat against the amount up to which we can tolerate mis-classifications in our training data.

By introducing Lagrange multipliers $\alpha_i, i = 1, \ldots, \ell$ the constrained problem (7) can be reformulated in the so-called dual form as follows:

$$\begin{aligned} &\text{Maximize } L_D \equiv \sum_i \alpha_i - \tfrac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \\ &\text{subject to } 0 \leqslant \alpha_i \leqslant C \\ &\qquad\qquad \sum_i \alpha_i y_i \mathbf{x}_i = 0 \end{aligned} \tag{8}$$

and the solution $\mathbf{w}$ can be computed as:

$$\mathbf{w} = \sum_{i=1}^{N_s} \alpha_i y_i \mathbf{x}_i \tag{9}$$

Note that, albeit there exist one α_i for each training data, only few (N_s) α_i will be greater than zero. Those points for which $\alpha_i > 0$ are called *support vectors* and lie on one of the two separating hyperplanes. Moreover, switching to Lagrange formulation allows the training data to appear only in the form of dot products between vectors. This is an interesting property that can be used to generalize the method in cases where we want the decision function to be a non-linear function of the data.

Suppose to map the data in some other Euclidean space H through the mapping $\Phi : \mathbb{R}^d \mapsto H$ in which the points are linearly separable. The method can be generalized in terms of *kernel function* by observing that is only required to define a kernel K such that $K(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)$.

Several kernels exist in literature with different characteristics. For our application we restricted to the evaluation of the linear kernel:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^{\mathrm{T}}\mathbf{x}_j \tag{10}$$

and the gaussian kernel:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \mathrm{e}^{-g||\mathbf{x}_i - \mathbf{x}_j||^2} \tag{11}$$

7.2 Cursor identity classifiers

To associate every device with its blob when *appear* and *stop* events are triggered, we trained two independent SVM-based classifiers. This choice is due to the fact that the feature set used in the first event is slightly different from the second. Indeed, if the accelerometer and compass data are always necessary to describe the device state in both events, the rate of growth of blob area is relevant only when a new blob appears and the blob velocity is only applicable just after a motion on the table.

Specifically, for the first classifier we collected vectors $\mathbf{x}_i \in \mathbb{R}^d$ composed by d distinct features. The first component of the vector is the angle difference in degrees between the orientation of the blob in the image space and the magnetic north measured by the device. Since a blob is seen as a rounded rectangle, only an undirected axis can be computed from its shape and so the difference is chosen as the minimum angle between the axis direction and the device orientation. It should be noted that just a rough orientation is required, thus small deformations of the observed shape (due to occlusion or to the hand holding the object) should have minimal influence. Once the number of samples s and history length h (in seconds) are chosen, the next $3s$ components of the vector $\mathbf{x}_i$ are just the concatenation of measured values of acceleration with respect to the three axis of the device. Each signal is linearly interpolated and re-sampled s times in the time span defined by h. Last s components of vector $\mathbf{x}_i$ are the measured values of blob area size in pixel, re-sampled in the time interval that spans from the first detection of the blob to the time in which the *appear* event is triggered.

In a similar way, for the second classifier we collected vectors whose first $3s + 1$ components are identical to the first case. Last $2s$ components are the concatenation of measured values of blob velocity computed by the table during its movement. Again, the measures are trimmed and re-sampled to match the corresponding acceleration signals.

Different values of s and h can be chosen to define the trade-off between the dimensionality of the vectors and the descriptiveness of the sensor data. Some possible combinations are proposed and evaluated in the experimental section.

7.3 Improved reliability via majority voting

The accuracy of the linear and Gaussian kernel-based classifiers for the *appear* and *stop* events is expected to be good enough to get a correct classification most of the time. However, given that several events could happen during the tracking lifespan of an object, a proper technique should

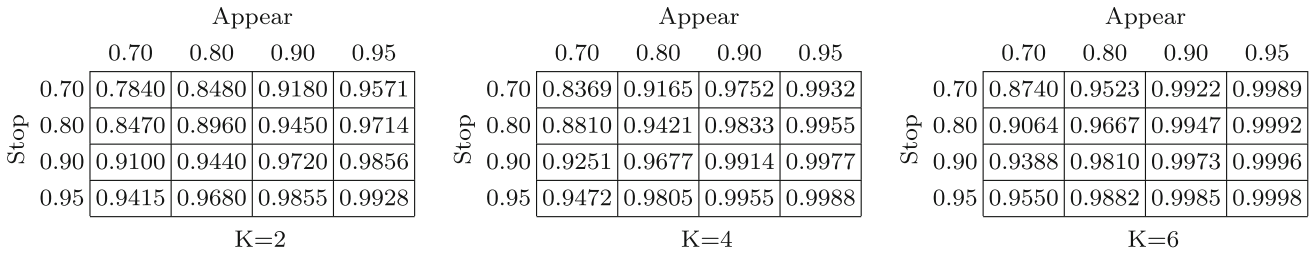


Fig. 11 Improvement in accuracy with sequences of stop events with different lengths

be adopted to get advantage of the added information supplied by subsequent classifications. To this end, we propose a very simple majority voting method where a blob that enters the tracking system is first identified through the *appear* classifier, then, if such blob remains consistently tracked by the camera, each possible *stop* event will cast an additional vote that can confirm or deny the initial recognition. Since, in our schema, we trust more the *appear* event, the initial identification is kept as valid if the votes against are not the strict majority. This method permits to correct initial association error and prevents further individual wrong classifications from compromising the correctness of the association.

It is interesting to analyze this approach from a probabilistic point of view in order to asses the improvements that are to be expected by applying such correction measure. First of all, we define with the symbol P_a the accuracy of the classifier of the *appear* event and with P_s that associated to the *stop* event. The probability to observe exactly i correct *stop* classifications over a total of k events can be computed according to the binomial distribution:

$$P_{\text{exact}}(k, i) = \binom{k}{i} P_s^i (1 - P_s)^{k-i} \tag{12}$$

Since all the $P_s(ki)$ are disjoint events, we can compute the probability of observing at least j correct classifications as:

$$P_{\text{atleast}}(k, j) = \sum_{i=j}^{i<k} \binom{k}{i} P_s^i (1 - P_s)^{k-i} \tag{13}$$

If we consider the first classification, we could have obtained a correct classification with probability P_a and a wrong one with probability $(1 - P_a)$.

Again, these events are clearly independent, thus the overall probability of getting a correct classification when applying majority voting after k observed *stop* events can be computed as:

$$P_a P_{\text{atleast}}\left(k, \left\lfloor \frac{k}{2} \right\rfloor\right) + (1 - P_a) P_{\text{atleast}}\left(k, \left\lceil \frac{k}{2} \right\rceil + 1\right) \tag{14}$$

In fact, at least $\lfloor \frac{k}{2} \rfloor$ correct *stop* event detections are needed to not spoil a good initial classification, while at least $\lceil \frac{k}{2} \rceil + 1$ are required to fix a wrong start.

In Fig. 11, we show the expected accuracy of a combined classifier, respectively, for two, four and six correction steps and with respect to a range of different accuracy for the base classifiers.

8 Experimental validation

The proposed approach has been tested by using it as a device identification for the multiuser map-based art browser described in Sect. 3. Two quantitative aspects of the system have been studied separately: the performance of the classifier (both with a linear and Gaussian kernel) and the accuracy of the time synchronization protocol used.

8.1 Classification performance

To assess the classification accuracy for both *appear* and *stop* classifiers a set of manually labelled data have first to be collected.

The positive examples (i.e. the ones for which the data refer to a correct device-blob association) are gathered by triggering *appear* and *stop* events keeping only one active device at a time. Data are recorded simulating a normal system usage with just a single device hence ensuring that the only blob visible will be the one produced by that device. Negative examples are collected in a similar way but exploiting inactive devices. Again, data are recorded during a normal system usage but placing on the table only inactive devices and using active devices on a fake non-interactive table. In this way, blobs seen by the table will never refer to their correct device.

We collected thousands of samples for *appear* and *stop* events to be used for training and testing. To compute the classifier accuracy with respect to linear and Gaussian kernels we performed a K-fold cross-validation to our data. In k-fold validation the data set is randomly partitioned into k subsamples. $k-1$ sub-samples are used to train the classifier and

the remaining sub-sample is used as the validation set to test the learned model. This process is repeated k times and the average accuracy is returned. This limits the over-fitting that may occur while learning the model and allows an effective exploration of method parameters. In all our experiments we chose $k = 5$.

For each of the two classifiers, linear and Gaussian kernels have been tested with three different type of data points, respectively, using a signal timespan of 1 s with ten samples, 0.5 s with ten samples and 0.5 s with five samples.

In Fig. 12 the accuracy of two classifiers with linear kernel is shown with respect to the parameter C. The classification performance of *appear* event is better than *stop*, probably for the lower dimensionality of the points that allow the data to be more linearly separable. In the first case the best accuracy is $\approx$97 % while considering a signal timespan of half a second. In the *stop* case the best accuracy is $\approx$93 % achieved with a timespan of 1 s. In both cases the value of C is not crucial to obtain good performances demonstrating that the training data are probably well separable into the two classes.

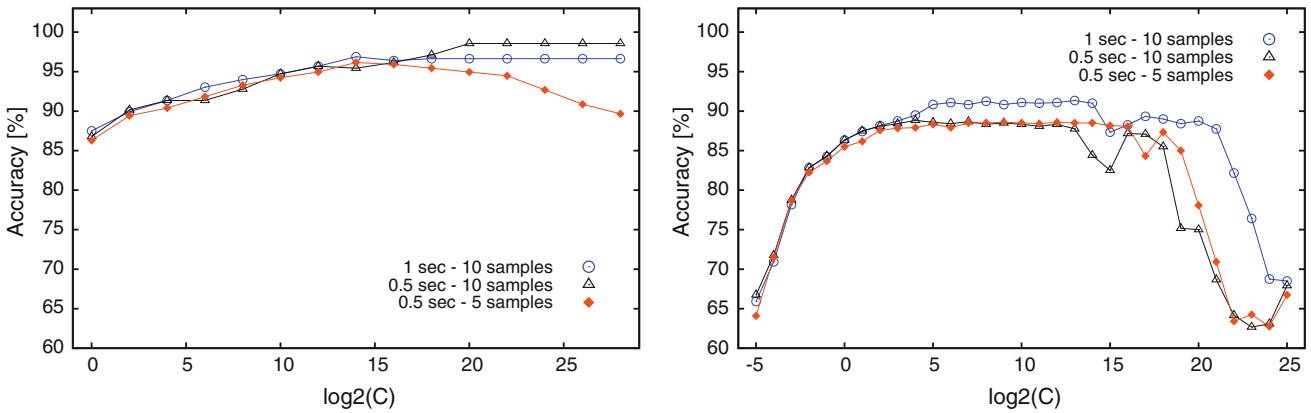


Fig. 12 Evaluation of the accuracy of the linear kernel SVM classifier for both the appear and stop events

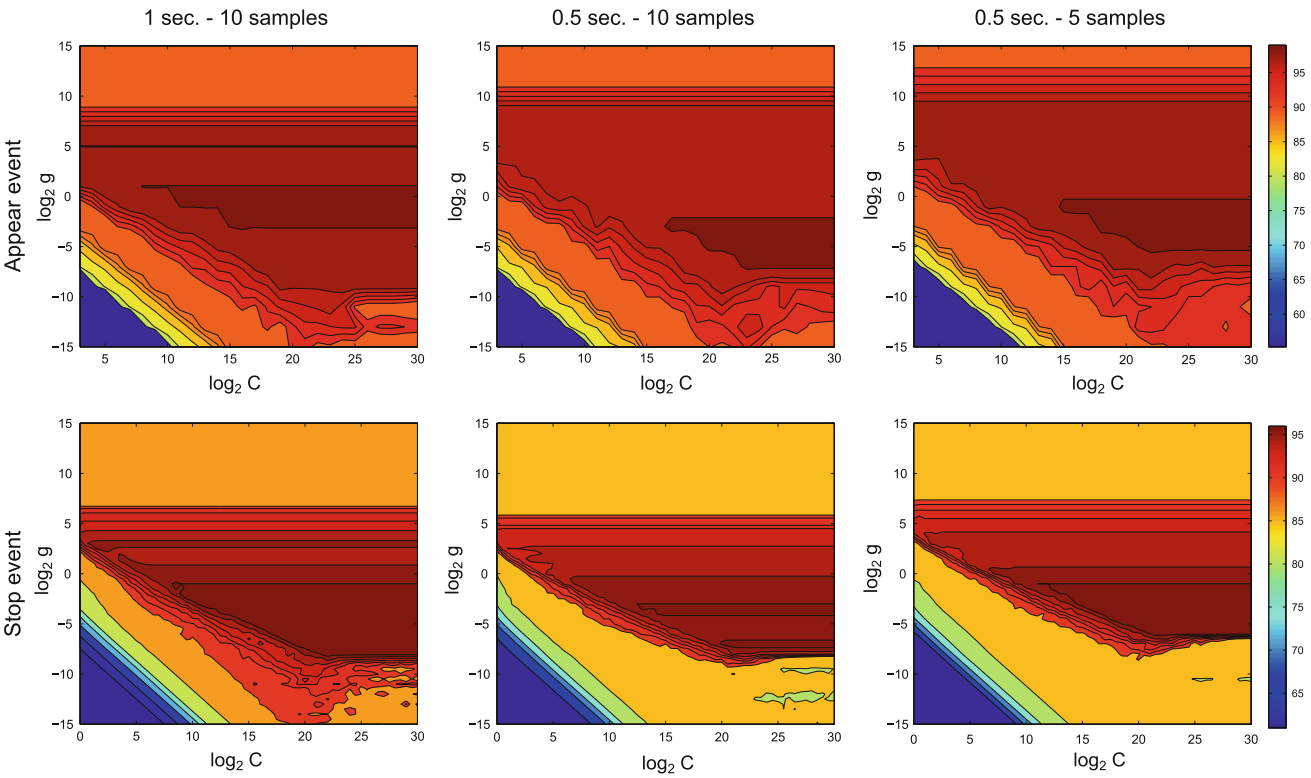


Fig. 13 Evaluation of the accuracy of the Gaussian kernel SVM classifier for the appear and stop events

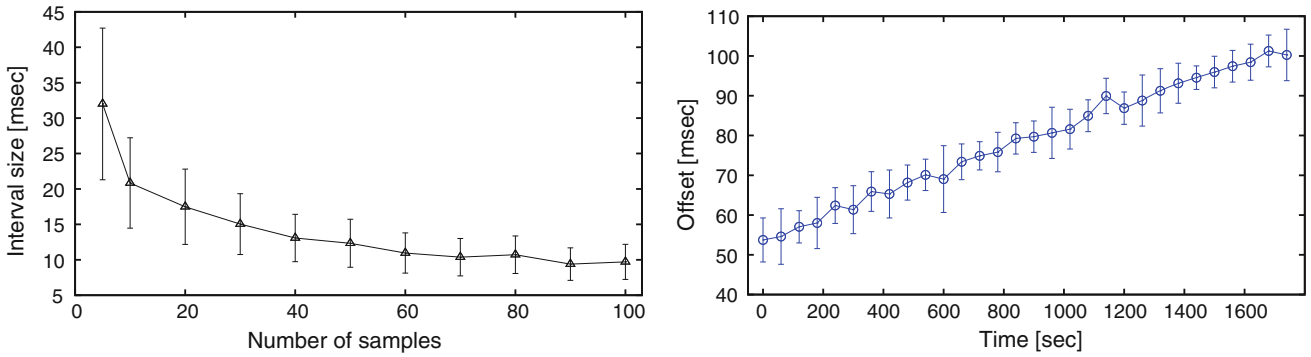


Fig. 14 Evaluation of the accuracy of the time synchronization with respect to the number of samples taken and of the amount of drift between the two real-time clocks with respect to the time elapsed from the last synchronization

In Fig. 13 the accuracy is examined as a function of C and g parameter space. Overall, the non-linear classifier obtains better performance with respect to linear case. Best accuracy of 99.3 % is obtained for the *appear* classifier with a timespan of 1 s with ten samples. However, it has to be noted that the portion of the C/g plane for which accuracy is above 95 % is wider for a timespan of 0.5 s. For the *stop* event, the accuracy is a little bit lower and the best performance is achieved for a timespan of 1 s, which is a behaviour similar to the one that has been found when dealing with the linear case.

This level of accuracy is already good enough to be used in many practical scenarios that are tolerant to a negligible degree of misclassification. However, it should be noted that, according with the probabilistic analysis done in Sect. 7.3, considering the obtained accuracies, the combination of the two analyzed classifiers could easily reach an extremely reliable recognition rate.

8.2 Time accuracy and drift

In Sect. 6.2, we described the technique adopted to synchronize the real time clock of the device with the time measured by the PC inside the table. In practice, this boils down to measure as precisely as possible the time offset between the two clocks.

In the left part of Fig. 14 we show the effect of the number of samples over the accuracy of the offset measure (i.e. the size of the intersection between all the measured intervals). It can be seen that after as few as 20 samples the accuracy is about 20 ms and seems to be asymptotically approaching 10 ms as the number of samples increases. An accuracy between 10ms and 20 ms is acceptable for our application since it is in the same order of the camera sampling, which happens at 30 fps (and thus, every 33 ms).

From a theoretical point of view, a large number of samples could easily be obtained by sending time synchronization messages regularly when the device is not transmitting other data. Unfortunately, in practice this is not possible because of the drifting between the two clocks, i.e. the slight but significant difference of the internal oscillators. The drifting between a device and the table has been measured using a sliding samples window. The resulting data are plotted in the right graph of Fig. 14. The drifting seems to be linear with the time (which is of course expected) and its value is in the order of about 50 ms over a time span of 20 min. This is indeed a large value and it implies that for the synchronization to give reasonable results the probing messages should be exchanged in a few seconds span. Further, given the sizeable drifting, synchronization should happen quite often.

9 Conclusions

We presented a blob identification approach that does not rely on fiducial markers or object tracking with external cameras. Rather, it takes advantage of the relation between sensor data gathered from the active device to be recognized and the behaviour of the blobs simultaneously observed by a camera. Such data are used to feed two SVM classifiers that can be optionally combined to obtain an improved accuracy level. Within the scope of this work, the proposed technique has been applied to an interactive table setup in which it was not possible to use classical marker-based systems. However, the overall approach is general enough to be suitable for many other scenarios where a mix of visual and sensor-based data can be exploited for markerless identification. Finally, the recognition accuracy that can be obtained has been assessed with an extensive set of experiments that explored the parameter space of two different kernel types.

Acknowledgments The *Venice Imago Project* is directed by Giuseppe Barbieri, Department of Philosophy and Cultural Heritage of Università Ca' Foscari Venezia, who is gratefully acknowledged for the provided support.

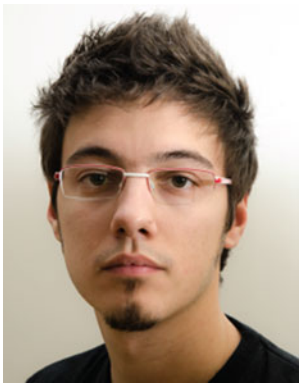
References

1. Ardito, C., Costabile M.F., Lanzilotti, R.: Gameplay on a multi-touch screen to foster learning about historical sites. In: AVI '10,

Proceedings of the International Conference on Advanced Visual Interfaces, pp. 75–78. ACM (2010)
2. Barbieri, G., Celentano, A.: Multimedia technology: a companion to art visitors. In: Koukopoulos, D., Styliaras, G. (eds.) Handbook of Research on Technologies and Cultural Heritage: Applications and Environments, chapter 19, pp. 393–410. IGI Global (2011)
3. Ben-David, A., Mandel, J.: Classification accuracy: Machine learning vs. explicit knowledge acquisition. Mach. Learn. **18**, 109–114 (1995)
4. Bergamasco, F., Albarelli, A., Torsello, A.: Pi-tag: a fast image-space marker design based on projective invariants. Mach. Vis. Appl. (2012)
5. Bergweiler, S., Deru, M., Porta, D.: Integrating a multitouch kiosk system with mobile devices and multimodal interaction. In: ACM International Conference on Interactive Tabletops and Surfaces, ITS '10, pp. 245–246. ACM, New York (2010)
6. Burges, C.J.C.: A tutorial on support vector machines for pattern recognition. Data Min. Knowl. Discov. **2**(2), 121–167 (1998)
7. Celentano, A., Orsini, R., Pittarello, F.: Towards an environment for designing and evaluating multimedia art guides. In: AVI '10, Proceedings of the Working Conference on Advanced Visual Interfaces, pp. 93–96. ACM (2010)
8. Cheshire, S., Baker, M.: Consistent overhead byte stuffing. IEEE/ACM Trans. Netw. **7**(2), 159–172 (1999)
9. Ciocca, G., Olivo, P., Schettini, R.: Browsing museum image collections on a multi-touch table. Inf. Syst. **37**(2), 169–182 (2012)
10. Cortes, C., Vapnik, V.: Support-vector networks. In: Machine Learning, pp. 273–297 (1995)
11. Dachselt, R., Buchholz, R.: Natural throw and tilt interaction between mobile phones and distant displays. In Proceedings of the 27th International Conference Extended Abstracts on Human Factors in Computing Systems, CHI EA '09, pp. 3253–3258. ACM, New York (2009)
12. Dietz, P., Leigh, D.: Diamondtouch: a multi-user touch technology. In: UIST '01, Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology, pp. 219–226. ACM (2001)
13. Dippon, A., Klinker, G.. Kinecttouch: accuracy test for a very low-cost 2.5D multitouch tracking system. In: Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces, ITS '11, pp. 49–52. ACM, New York (2011)
14. Dohse, T., Still, J.D., Parkhurst, D.J.: Enhancing multi-user interaction with multi-touch tabletop displays using hand tracking. In: First International Conference on Advances in Computer–Human, Interaction, pp. 297–302 (2008)
15. Döring, T., Shirazi, A.S., Schmidt, A.: Exploring gesture-based interaction techniques in multi-display environments with mobile phones and a multi-touch table. In: Proceedings of the PPD '10, Workshop on Coupled Display Visual Interfaces, in Conjunction with AVI 2010, pp. 47–54 (2010)
16. Echtler, F., Nestler, S., Dippon, A., Klinker, G.: Supporting casual interactions between board games on public tabletop displays and mobile devices. Personal Ubiquitous Comput. **13**, 609–617 (2009)
17. Han, J.Y.: Low-cost multi-touch sensing through frustrated total internal reflection. In: UIST '05, Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology, pp. 115–118. ACM (2005)
18. He, Z.: Accelerometer based gesture recognition using fusion features and SVM. J. Softw. **6**(6), 1042–1049 (2011)
19. Hesselmann, T., Henze, N., Boll, S.: Flashlight: optical communication between mobile phones and interactive tabletops. In: ACM International Conference on Interactive Tabletops and Surfaces, ITS '10, pp. 135–138. ACM, New York (2010)
20. Hornecker, E.: "I don't understand it either, but it is cool"—visitor interactions with a multi-touch table in a museum. In: TABLETOP 2008. 3rd IEEE International Workshop on Horizontal Interactive Human Computer Systems, pp. 113–120 (2008)
21. Jacucci, G., Morrison, A., Richard, G.T., Kleimola, J., Peltonen, P., Parisi, L., Laitinen, T.: Worlds of information: designing for engagement at a public multi-touch display. In: CHI '10, Proceedings of the 28th International Conference on Human Factors in Computing Systems, pp. 2267–2276. ACM (2010)
22. Kato, H., Billinghurst, M.: Marker tracking and HMD calibration for a video-based augmented reality conferencing system. In: Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality. IEEE Computer Society, Washington, DC (1999)
23. Ketabdar, H., Jahanbekam, A., Yuksel, K.A., Hirsch, T., Abolhassani, A.H.: Magimusic: using embedded compass (magnetic) sensor for touch-less gesture based interaction with digital music instruments in mobile devices. In: Proceedings of the Fifth International Conference on Tangible, Embedded, and Embodied Interaction, TEI '11, pp. 241–244. ACM, New York (2011)
24. Knecht K., Konig, R.: Augmented urban model: Bridging the gap between virtual and physical models to support urban design. In: CONVR 2011: Proceedings of the 11th International Conference on Construction Applications of Virtual Reality, pp. 142–152 (2011)
25. Marquardt, N., Kiemer, J., Greenberg, S.: What caused that touch?: expressive interaction with a surface through fiduciary-tagged gloves. In: ACM International Conference on Interactive Tabletops and Surfaces, ITS '10, pp. 139–142. ACM, New York (2010)
26. Marzullo. K.A.: Maintaining the time in a distributed system: an example of a loosely-coupled distributed service. PhD thesis, Stanford (1984)
27. Multimedia Information & Interaction Laboratory, Università Ca' Foscari Venezia. Interactive Multimedia Art Guide Project. http://www.dais.unive.it/auce/artguide
28. Boorsch, S.: Six Centuries of Master Prints. Cincinnati Art Museum, Cincinnati (1993)
29. Schöning, J., Rohs, M., Krüger, A.: Using mobile phones to spontaneously authenticate and interact with multi-touch surfaces. In Proceedings of PPD 2008, Workshop on Designing Multi-touch Interaction Techniques for Coupled Public and Private, Displays (2008)
30. Sehgal A.K., Das, S., Noto, K., Saier, M., Elkan, C.: Identifying relevant data for a biological database: Handcrafted rules versus machine learning. IEEE/ACM Trans. Comput. Biol. Bioinform. **8**(3), 851–857 (2011)
31. Shirazi, A.S., Winkler, C., Schmidt, A.: Flashlight interaction: a study on mobile phone interaction techniques with large displays. In Proceedings of the 11th International Conference on Human–Computer Interaction with Mobile Devices and Services, MobileHCI '09, pp. 93:1–93:2. ACM, New York (2009)
32. Sun, L., Zhang, D., Li, B., Guo, B., Li, S.: Activity recognition on an accelerometer embedded mobile phone with varying positions and orientations. In: Ubiquitous Intelligence and Computing, pp. 548–562 (2010)
33. Villanueva, P.G., Gallud, J.A., Tesoriero, R.: WallShare: a collaborative multi-pointer system for portable devices. In: Proceedings of PPD 2010, Workshop on Designing Multi-touch Interaction Techniques for Coupled Public and Private Displays, pp. 31–34. ACM (2010)
34. Wagner, D., Reitmayr, G., Mulloni, A., Drummond, T., Schmalstieg, D.: Real time detection and tracking for augmented reality on mobile phones. In: IEEE Transactions on Visualization and Computer Graphics, vol. 99 (2010)
35. Wu, J., Pan, G., Zhang, D., Qi, G., Li, S.: Gesture recognition with a 3-d accelerometer. In: Proceedings of the 6th International Conference on Ubiquitous Intelligence and Computing, UIC '09, pp. 25–38. Springer, Berlin (2009)

Author Biographies

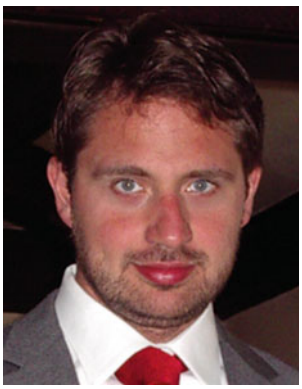
Andrea Albarelli worked in the industry for 10 years before receiving his Ph.D. in Computer Science from the University Ca' Foscari Venice in 2010. He has taught Computer Architecture, Information Theory and Computer Vision. He co-authored more than 40 peer-reviewer technical papers, mainly in the field of Computer Vision, with particular attention to issues of representation and processing of 3D data and the adoption of Game Theory in the context of Pattern Recognition problems, ranging from point-pattern matching to surface registration. He has participated in several research and technology transfer projects funded by both public and private partners. Since 2010, he is a founding member of an academic spin-off acting as a bridge between industry and Computer Vision research. In 2010 he was the winner of the tender "IMPRESA", sponsored by the Ministry of Economic Development, the prize "Working Capital" for innovative young researchers and the "NVIDIA Best Paper Award". In 2011 he won the "Award for Research" established by Venice University to reward young researchers exhibiting the highest scientific impact.



Filippo Bergamasco received an MSc degree (with honors) in Computer Science from Ca'Foscari University of Venice, Venice, Italy, in 2011 and is currently a PhD candidate at the University Of Venice. His research interests are in the area of computer vision, spreading from 3D reconstruction, Game-Theoretical approaches for matching and clustering, structure from motion, augmented reality and photogrammetry. He has been involved in many commercial computer vision projects for industry and entertainment, including structured light scanner solutions, pipes measurement system for automotive, interactive vision-based museum exhibitions and AR applications for embedded devices.

Augusto Celentano is full professor of Information Systems at Ca' Foscari University Venice, Italy. He received a Master Degree in Electronic Engineering from Politecnico di Milano in 1973. He has been the Deputy Rector for information technology and Head of the Department of Computer Science of Ca' Foscari University Venice. Augusto Celentano has been a member of scientific committees in research and educational centers of Politecnico di Milano and Ca' Foscari University Venice, and has been consultant in research projects of the European Union. His current research interests are in interactive multimedia systems, Human Computer Interaction and Digital Humanities. He is the co-author of more than 90 scientific papers published in international journals and conference proceedings, and is serving as chair and program committee member in several international conferences.

Luca Cosmo received an MSc degree with honors in Computer Science from the University of Venice, Italy, in 2012. He is currently a PhD candidate at the University of Venice where his field of research is the non-rigid matching. He has been involved in computer vision projects for industry and museum exhibitions. For hobby he deals with the development of the graphical engine and networking of PC strategy games.



Andrea Torsello received his PhD in computer Science at the University of York, UK and is currently working as Assistant Professor at University Ca Foscari Venice, Italy. His research interests are in the areas of computer vision and pattern recognition, in particular, the interplay between stochastic and structural approaches as well as Game-Theoretic models, with applications in 3D reconstruction and recognition. Dr. Torsello has published more than 80 technical papers in refereed journals and conference proceedings and has been in the program committees of numerous international conferences and workshops. In 2011 he has been recognized as "Distinguished alumnus" by the University of York, UK. He held the position of chairman and is currently the vice-chair of the Technical Committee 15 of the International Association for Pattern Recognition, a technical committee devoted to the promotion of research on Graph-based Representations in Pattern Recognition.