Nomadic gestures: A technique for reusing gesture commands for frequent ambient interactions

Radu-Daniel Vatavu
University Stefan cel Mare of Suceava, 13 Universitatii, 720229 Suceava, Romania
E-mail: vatavu@eed.usv.ro

Abstract. The age of ambient intelligence has already incorporated gestures into practical applications with the goal of delivering adaptive and personalized interactions. However, practitioners are faced with many problems when implementing gesture-based interfaces for such interactive ambient systems. Although gestures offer great opportunity for natural and intuitive interactions, there are currently little to no rules for creating the set of gestures for a given application. Therefore, designers associate gestures and functions by relying only on their own expertise and experience which leads to different systems exposing different standards. This approach does not only create the premises for confusion among users in a future world with such hundreds daily interactions but it also contradicts the goals of ambient intelligence in which interaction should be personalized and adapted to each user. This work introduces a novel concept (nomadic gestures) for reusing a set of user-defined gesture commands in the context of interacting with ambient systems. Nomadic gestures live on each user’s personal mobile device and are uploaded to the ambient system prior to interaction. The concept relies on an important shift of perspective strictly adhering to the goals of ambient intelligence: it is not the users that adapt to the interface learning its commands but instead the interface employs the users’ own gesture sets with their own preferred function associations.

Keywords: Ambient displays, ambient intelligence, gestures, gesture-based interfaces, mobile phones, ambient interactions, personalized and adapted interaction, computer vision, kinect

1. Introduction

Gestures represent powerful and intuitive means to point, manipulate objects, and transmit meaningful information, while they also add multimodality to one’s self-expression when accompanying speech [22,32]. Therefore, gestures have intrigued researchers since the early days of computing [8,52] for their potential of supporting fluent, intuitive, and natural interactions. The latest advances in acquisition and processing technologies [11,18,19,29,50] have finally made gesture-based interfaces affordable and accessible and, consequently, gestures have found their way into consumer products and applications. As a result, gestures are now common for interacting with mobile devices [61]; playing computer games in which the user’s body movements are mapped to the motions of virtual characters [13,40]; working with touch-sensitive surfaces [20]; and pointing to and interacting with large ambient displays [5,7,12]. The strong interest showed by researchers for investigating gestures as well as by designers and practitioners that include gestures in their applications is motivated by the convenient attributes gestures naturally possess: familiarity and intuitiveness.

Ideally, gestures bring a considerable promise for user interfaces: the promise of natural interaction. In this ideal scenario, users would find gesture-based interfaces easy to understand and use and therefore easy to master. Although current research and practice show that the community is still far from this ideal scenario as noted by experts in interaction design [28,38,39], gestures have been found to work outstanding for computer games [23,43,58]; entering fast pen strokes [26,27,60]; as well as for acting as shortcut commands.
[3,30] leading to efficient small task completion times for users having attained expert performance.

However, when moving beyond touch-screen mobile interfaces or computer games with somewhat fixed constraints, implementing gesture-based interaction efficiently and effectively in a non-constrained ambient scenario represents a considerable challenge. And this goes especially for the not-so-far-away computing age in which prospective users would need to migrate from one interactive ambient system to the next as per the vision of ubiquitous computing and smart environments. Therefore, in order to accomplish such a challenging goal, researchers in ambient intelligence (AmI) have been investigating new ways for detecting, recognizing, and understanding human gestures [50,55] but also prospecting practical application opportunities for AmI scenarios [14,44]. The end of using gestures for ambient interactions is essentially the same one that drives gesture research in the human-computer interaction (HCI) community: users or customers of AmI services would interact with ambient systems in a natural, intuitive, and familiar manner without spending cognitive effort for learning new ambient interfaces. However, when trying to put gestures into practice, the notions of natural and intuitive seem to puzzle designers and practitioners. Although gestures bring many advantages, the designers of such interfaces are faced with considerable problems. Let alone the complexity of acquisition algorithms as well as the processing power required by real-time non-invasive recognition of free-hand and whole-body gestures [33,45], special care needs to be devoted to designing the application gesture set. The main problems consist in designing gestures that prove easy to execute, learn, and recall [37] without which no satisfying user experience can be created.

Another more recent problem comes to add further difficulty to the already existing challenges: the proliferation of both gesture-based interfaces and interactive ambient systems. As ambient intelligence develops, more and more systems will presumably make use of gesture commands, leading to a world in which gesture-based interactions will be prevalent. However, as noticed by experts [38,39], today’s designers of gesture-based interfaces either neglect decades of HCI usability practices or are totally excited and therefore distracted by advances in technology that they forget basic usability principles when implementing gesture-based interfaces. For example, different gestures are being proposed for the same function by each application designer because, in the absence of a standard, the composition of the gesture set is at the mercy of the designer’s resolution. As a consequence, users will find difficult to recall gesture commands while they move from one ambient interactive system to the next which will certainly create confusion. Even more, even if such a standard would eventually be developed, different users still prefer different gesture commands for the same function. This has already been observed by researchers when eliciting gesture commands from non-technical users [34,48,59] with the puzzling effect that users also seem to prefer different gestures than the ones expert designers are proposing [34,59]. The problem is especially important in the context and vision of gesture-controlled ambient displays for which the interaction is usually brief, casual, and time- and location-opportunistic which gives no time for users to explore, learn, or practice the interface before using it.

1.1. Contribution

Rapidly moving into a world in which ambient systems will likely consider gesture-based interactions exclusively may cause usability problems. The main concerns come from the variability of gesture sets proposed by different application designers but also from the variability in users’ preferences for gesture commands that contradicts the adoption of a standard. By constraining users to learn gestures they may not necessarily find intuitive or natural or by allowing systems to expose different gestures for the same functions will surely confuse users and affect negatively the perception of a natural interaction experience.

The article introduces the concept of user-defined gestures that can be reused for new and unseen ambient displays especially in unknown environments (that users never accessed before, a likely scenario in a future world of prevalent interactive ambient systems). We denote the concept of reusing gestures across many systems as nomadic gestures. The idea is to have users carry their preferred and practiced gesture sets on some storage device (the mobile phone being a good example) and upload them to the public ambient display they are about to interact with. The interface will then associate the users’ own gestures with the tasks available in the system. The advantages are two-fold:

(i) The robustness and accuracy of gesture recognition algorithms will considerably improve (as systems will employ the users’ own executions as training examples acting as if user-specific training had just occurred);
(ii) Users will already know how to interact with the system by reusing their own gestures for executing commands for which they have already created solid mental associations.

The article introduces the concept of nomadic gestures and discusses a case study employing free-hand motion gestures. The goal behind nomadic gestures is to promote reuse of user-specific gesture sets when encountering unknown ambient interfaces building thus on two of the goals of ambient intelligence: designing adaptive and personalized systems and technology as per the goals set by Aarts et al. [1,2].

1.2. Article organization

The article is organized as follows. Section 2 conducts an overview of related works that investigate techniques for interacting with public ambient displays by considering the two most frequent technologies being proposed today: mobile phones and gesture recognition. An overview of the principles and guidelines of designing gesture sets in the HCI community is also presented. The limitations and challenges of these interaction techniques and design principles are being highlighted. Section 3 introduces the concept of nomadic gestures by setting design requirements and discussing usage scenarios. Section 4 describes implementation details of the concept in the form of a Music Video Shop ambient display. The article concludes by presenting the details of an informal user study conducted in order to inform new designs and future development of the nomadic gestures concept.

2. Related work

We focus on related works that address the problem of interacting with public ambient displays in order to highlight the common design practices as well as to identify the problems designers are currently facing. We discuss the technologies frequently proposed for interacting with public ambient displays which include mobile phones [5,7,12] and gesture recognition [16,24,31,53,57].

2.1. Mobile phones and gesture-based interaction techniques for ambient displays

Mobile phones have been primarily used for controlling cursors on remote displays in a WIMP-like fashion [5,7,12] but also as sensing devices for capturing gestures in the form of tilting and throwing [12]. Researchers have been investigating mobile phones due to their many advantages to indirectly control a remote display: immediate-availability of sensing and processing power; familiarity as they are personal devices subjected to prolonged usage; a nice complementarity between the mobility of the device and the large screen area of the ambient display.

Beyond mobile devices, researchers have explored various technologies in order to capture and use gestures for interacting with ambient displays. Implementing a successful gesture-based interface requires carefully selecting a technology for acquiring gestures, implementing a recognizer, and designing a set of gesture commands. Each of these aspects brings in considerable challenges that relate to the robustness of acquisition/recognition and efficiency/effectiveness of the interaction [54]. Vogel and Balakrishnan [57] explored the use of free-hand gestures for which they proposed several pointing and clicking techniques. Malik et al. [31] used computer vision in order to capture and recognize the postures of the user's hands above the surface of a table. Shoemaker et al. [53] introduced the shadow reaching technique which uses a perspective projection of the user's shadow on the remote display for providing easy access over the entire area of the display. Although selection has usually been implemented using pointing [57], non-pointing acquisition of remote targets that make use of gestures have recently been proposed [10,15]. Pan and zoom techniques for remote displays have been investigated by Nancel et al. [36] which compared single and two-hand interaction, linear and circular movements, and explored the use of different guidance levels for gestures performed in mid-air. Also, we note the different acquisition technologies employed today by practitioners implementing gesture recognition: vision [31,46], dedicated motion sensors [57], acceleration data and gloves [16]. For example, Huang et al. [16] explored different techniques for recognizing gestures captured from a data glove. Krishnan et al. [25] investigated activity recognition using wearable accelerometers.

The research literature is divided between using the mobile phone or gestures when proposing interaction techniques for remote displays. Each solution has its strengths and weak points and therefore mixed designs have been considered as well [12]. The concept introduced by this work also considers a hybrid approach that combines the always-availability of storage and communication power of mobile devices and natural-
ness of free-hand gesture-based interaction. A stronger comparison between the two approaches is conducted in a later section while discussing the design requirements that led to the concept of nomadic gestures.

2.2. Designing gesture commands

Although many studies have been devoted to developing new acquisition, recognition, and interaction techniques, few have explored the design of the gesture set. The current practice lets designers propose gesture commands based on their previous expertise and experience. Beyond that, there are a few works that provide techniques to assist users at learning gesture commands such as the GestureBar control of Bradford et al. [9] or the OctoPocus dynamic guide of Bau and Mackay [6]. However, these techniques only assist users in transitioning from novice to expert modes but this still implies that users need to learn new gestures before they start using them.

Recent studies have shown that different users prefer different gestures for the same function. For example, Wobbrock et al. [59] elicited tabletop gestures from non-technical users by portraying the effect of a command and asking for the gesture that would trigger the portrayed effect. They found that users rarely care about the number of fingers they use; that one and two-hand gestures can be used for the same command; and that there are commands for which little agreement can be observed between users. A subsequent study of Morris et al. [34] showed that users prefer gestures considerably different than those proposed by experienced designers. This suggests the need for personalization and adaptation in gesture-based interfaces.

An important design consideration when building a gesture-based interface is to accommodate for input variability: the same gesture will be executed differently each time (not just by different users but also by the same user at different moments of time). The impact of this observation is that gesture recognizers which are trained per user (user-specific training) perform much better than those trained for user-independent scenarios [42].

As a common practice to alleviate the above mentioned problems, current gesture-based interfaces allow users to enter their own gesture examples as well as to define their custom gestures for different functions. For example, Mozilla Gestures Redox1 allows Firefox users to perform navigation tasks quicker by using gesture commands. Users have the option to personalize their gestures or to add new ones. A common repository on the add-on website is even provided for users to share and exchange their custom gestures2. While simple in terms of recognition (detecting the stroke direction) and acquisition technology (mouse), Mozilla Gestures exemplifies perfectly the concept of user personalization of gesture-based interfaces. Following the same idea, the iGesture framework3 and the java toolkit of Appert and Zhai [3] allow the same personalization option. The trend of user-defined gestures can also be observed with more complex acquisition technologies such as tabletops and interactive surfaces that provide users with many opportunities to specify and enter commands (e.g. the same gesture of zooming can be executed using one hand with two fingers, one hand with many fingers, or two hands). Therefore, a growing interest has been manifested recently in allowing users to customize their gesture sets.

3. Nomadic gestures: A concept for reusing gesture commands across different interactive systems

The discussion from the previous sections has identified at least two common problems encountered when implementing gesture-based interfaces:

(i) The designer’s association between gestures and functions is not always intuitive to everyone [38] and different users will prefer different gestures for the same action. This has been clearly showed by Wobbrock et al. [59], Morris et al. [34], and Ruiz et al. [48] in their experiments of eliciting users’ gesture preferences;

(ii) User-dependent is preferred to user-independent training as gesture recognizers will perform more accurately [42]. However, going through a training stage each time when interacting with a new system pose problems with respect to the fluidity of the interaction. This becomes particularly important when on-the-run consumers need to perform hundreds of daily interactions in the future vision of interactive ambient systems becoming prevalent.

---

1Mouse Gestures Redox Home, http://www.mousegestures.org/

2http://www.mousegestures.org/exchange/

3http://www.igesture.org/index.html
The two most frequent approaches for interacting with public ambient displays, phone and gesture, each possess both strong and weak points. Therefore, proposing a hybrid approach which would combine both the storage and computing power of mobile phones and naturalness of gestures may serve well for addressing the above mentioned problems.

3.1. Mobile devices vs. free-hand gestures

As discussed in the related work section, previous solutions that have been proposed for interacting with remote displays either use mobile phones or gesture recognition. The proposed concept is actually a mix of the two: the mobile phone represents the storage medium for user-specific data (i.e. the gesture set) while the remote display implements and runs the gesture recognizer. Therefore, an in-depth analysis on the similarities and differences between these two major approaches is mandatory as each approach presents both advantages but also weaknesses for practitioners that build such systems but also for users that employ them. The two approaches are not exclusive and they may overlap in some designs such as the case of using tilt or throw movements [7,12] captured using the sensing capabilities of the mobile phone.

Table 1 presents a deeper analysis at different levels for the two approaches. The strong points relate to familiarity and personal aspects which are common for both mobile phones and gestures and which may act as strong factors in users preferring one over the other. For example, mobile phones represent personal equipments to which a special connection develops over time and for which the interface is completely understood due to long and continuous usage. On the other hand, gestures are familiar as they are used in every day interactions in the real world but also extremely personal as they show individuality and reveal personality. The two approaches as implemented today also share weak points that still need to be addressed carefully by future designs such as low interaction fluidity, privacy concerns, and low social acceptance. For example, restrictions exist today in using mobile phones in some public contexts. Also, recent studies have showed that people may not be willing to use gesture commands in public in some locations or in front of an audience [47].

3.2. Design requirements

Table 1 identifies the familiarity and personal levels as important elements shared by both approaches that should be exploited together. Also, mobile phones possess connectivity, sensing, and storage capacities while gestures bring in naturalness and intuitivity to the interaction. Good designs and smart sensing (see the discussion of the informal study later in the paper) may alleviate the problem of interaction fluidity while studies are still needed in order to better understand the social acceptance and privacy concerns when interacting using gestures in public settings.

By analyzing the challenges for interacting in unconstrained and unknown ambient environments and by considering the above mentioned factors for the two approaches, several requirements have been identified as key elements for the design of the new concept:

1) User-dependent training: Gesture recognizers should work if possible with user-dependent training data so that they would reach peak performance. This will have the effect of users perceiving the interface as reliable and robust;

2) User-dependent gestures: The interactive systems should accept different gestures (user-dependent) for the same function. This addresses the problem of gesture preference observed in the literature but also the intuitivity of the interaction: by using gestures that are perceived as valid or right for a given task, the overall feeling is one of intuitive and natural interaction as gestures and functions match perfectly as per the user’s mental model;

3) Short, to-the-point, and casual interactions: Considering the constraints of short, casual, and to-the-point interactions with ambient displays (see the introductory section), no time is available for users to train each interactive system with their own executions. Therefore, user-dependent training should take other forms than explicit entering of gesture executions.

These three design requirements relate directly to the goals of ambient intelligence consisting in developing systems and technologies that should be personalized and adaptive, two of the five goals articulated by Aarts et al. [1,2] next to embedded, context aware, and anticipatory technologies.

3.3. Nomadic gestures

Based on the design requirements, a new concept is proposed which consists in having users employ their preferred gesture set each time when interacting with a public ambient display. The preferred gestures are be-
Table 1
Strong points and challenges for mobile devices vs. gesture-based interaction with public ambient displays today

<table>
<thead>
<tr>
<th>Mobile devices</th>
<th>Gestures</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Strong points</strong></td>
<td><strong>Challenges</strong></td>
</tr>
<tr>
<td>1. <strong>Familiarity</strong>: interaction techniques and interfaces (buttons, stylus, and touch) of mobile phones are familiar to their users due to continuous practice and every day usage;</td>
<td>1. <strong>Fluidity of the interaction</strong>: in order to transmit content, mobile phones require a connection to be established (either by wireless LAN, Bluetooth, or IR) which may take time and therefore affect the fluidity of the interaction. This also relates to serendipity which means the users’ ability to spontaneously interact with the ambient display;</td>
</tr>
<tr>
<td>2. <strong>Personal</strong>: mobile phones represent personal belongings that connect to their owners on multiple levels such as pleasure of use, accustomedness and familiarization, pride in brand ownership, and emotional attachment. All these could translate into users preferring the mobile phone as a personal interface for interacting in (unknown) ambient environments;</td>
<td>2. <strong>Additional dependence</strong>: may need downloading and installation of additional software that would enable the interaction;</td>
</tr>
<tr>
<td>3. <strong>Connectedness</strong>: all new-generation smart phones present multiple options for connecting to networks or exchanging data with other devices such as Infrared, BlueTooth, and wireless network cards;</td>
<td>3. <strong>Privacy concerns</strong>: related to personal data stored on mobile phones as well as to possible GPS location tracking;</td>
</tr>
<tr>
<td>4. <strong>Sensing</strong>: most mobile phones include some kind of sensing equipment such as accelerometers and orientation sensors, video cameras, touch screens, light and proximity sensors;</td>
<td>4. <strong>Social acceptance</strong>: there are restrictions for using mobile phones in some public contexts.</td>
</tr>
<tr>
<td>1. <strong>Familiarity</strong>: gestures are being used and practiced every day in real world settings when interacting with real objects, pointing, or conveying additional meaning to speech utterances;</td>
<td>1. <strong>Fluidity of the interaction</strong>: reliable and robust tracking and recognition technology are mandatory for detecting users and their actions. False positives (valid gestures not recognized) and false negatives (noise motions falsely detected as valid gestures) demand either undo actions or reentering of commands with direct impact on the fluidity of the interaction;</td>
</tr>
<tr>
<td>2. <strong>Personal</strong>: gestures are extremely personal in the measure that they can identify individuals with unique motion and behavioral patterns. They are used to express emotion states, transmit additional meanings in communication, and to reflect personality;</td>
<td>2. <strong>Privacy concerns</strong>: that may relate to the face or user’s actions which can be captured by video cameras (and eventually stored);</td>
</tr>
<tr>
<td>3. <strong>Intuitivity and naturalness</strong>: gestures are intuitive to perform and are used naturally in all life situations. They are so natural that people use gestures even when speaking on the phone and blind people gesture as they speak just as much as sighted individuals do;</td>
<td>4. <strong>Social acceptance</strong>: low willingness of performing gestures in public has been observed. Willingness to interact with gestures depends on location and audience.</td>
</tr>
<tr>
<td>4. <strong>Tangible feedback</strong>: Touch-sensitive and tangible interfaces give immediate feedback to users touching their interactive surfaces which enhances the user experience of the interaction. Also, haptic feedback can be provided by modulating frequency responses in materials touched by fingers;</td>
<td></td>
</tr>
</tbody>
</table>

Also, the solution takes advantage of the strong points of both mobile phone and gesture approaches: storage and connectivity features of the mobile phones complemented with the naturalness and intuitivity of interacting with one’s own preferred gestures. The gesture set can be stored on the mobile phone in the form of storing stored on the mobile device and uploaded to the interactive system before interaction takes place. This solves gracefully the problems mentioned in the introductory sections, ensuring a perfect match between gesture and function for each user as well as robust recognizer performance due to user-specific training.
of a data file containing the association between gestures and functionality. Users connect to the public display via the available wireless connection of the device (Bluetooth, wireless LAN, etc.) and simply transfer the data file to the ambient display. Therefore, nomadic gestures touch on all the strong points of both approaches as discussed in Table 1. Figure 1 illustrates the concept.

It is important to note that the gesture configuration file must already exist which implies that a training procedure already took place. This can happen at the user’s pace and it can be more than a one-time procedure: it should continue each time new gestures are being added for new functions or when existing gestures are being removed or updated. This should be viewed as the user-specific training that takes place at the user’s own rhythm independently from the actual interaction in the ambient environment.

3.4. Examples of interactive scenarios for nomadic gestures

In order to develop and discuss further the concept of nomadic gestures, a few usage scenarios are being described in the following for which the advantages of the new concept are being highlighted.

3.4.1. Scenario case A: The railway station

Bill is a frequent traveler by train, as he commutes weekly to different cities. The railway stations he connects to dispose of large ambient displays that show train departure schedules. Sometimes, the information that Bill receives from these displays is sufficient in order to get oriented but sometimes he needs more content or needs the information fast hence interaction with such displays is mandatory. Also, as he is usually in a hurry, he doesn’t have the necessary time to explore the interface of such displays that are often new to him (as he visits some cities for the first time).

On the other hand, Bill is very much accustomed with his personal mobile device that allows touch-based interaction and that he uses to manage everything from contacts to his personal schedule. As most users of new generation touch-sensitive devices, he is very fond of touch gestures such as swipe, tap, and pinch. He already knows how to interact using these gestures which he finds familiar and natural. Over sustained usage, Bill has formed a solid mental model in which these touch gestures are associated to various functions. The ideal interface for Bill is therefore a touch-based one however the available system (the interactive ambient display) is evidently remote and out of reach. Therefore, he uses his mobile device in order to connect to the ambient display after which he uploads the gesture commands to the display terminal (Fig. 2A). Afterwards, he selects his train and asks for more details by making use of his preferred and familiar tap and swipe gestures.

3.4.2. Scenario case B: Window shopping

Jane goes shopping often and each time she passes by one of her favorite clothes store she likes to browse the contents of the store using the ambient display located inside its street window (Fig. 2B). This way, she can easily find out what the new additions are and which items are on sale. However, there is no time for learning how to use the gesture-based interface of each display and remembering multiple commands is quite a challenge. Instead, Jane uses the mobile phone in order to upload her gesture preferences to the ambient display consisting in commands frequently used such as next, previous, expand, and back. After which, she moves her hands in front of the display and immediately starts browsing through the store products.

3.4.3. Scenario case C: The video game store

Andrew, a 13 years old teenager, is entering his local video games store. He stops in front of a display containing information on the best selling new games. As he enjoys playing Kinect [23], he is expert at using free-hand and whole body gesture-based interfaces. He uploads his gesture motions and starts using them: he shakes his hand in front of the device to activate; flicks his hand from left to right to browse through the collection of video games; he then performs a special secret gesture in order to find the latest additions for his favorite game. His secret gesture involves whole body movements just like his character would do inside the game, a gesture he is fond of performing (as the gesture connects him directly to the game character).
3.4.4. Scenarios discussion

It must be noted that each person from each case study has a preferred acquisition technology besides a custom gesture set, e.g., the touch-based interface of the mobile phone or the free-hand gestures. Nevertheless, the concept still applies independently of the acquisition device as long as the requirements stay the same: the gesture set is transferred to the ambient display prior to interaction.

As already discussed, user data contains both the gesture set (that would ensure user-dependent training) but also gesture associations to system functions. For example, in the 1st scenario, Bill defines two gesture commands: a *swipe* gesture for browsing through the list of departing trains and a *tap* gesture for learning more information about a selected target. Also, he may define a custom gesture such as an *L* for directly inquiring the information about the next train going to Lisbon. This type of letter recognition for mobile phones has been found to work very well for searching instead of browsing the ever-increasing amount of data items being stored on the mobile device [30]. For the 2nd scenario, Jane only needs to remember and use 4 gestures (and 4 functions respectively): *next* and *previous* for browsing in an ordered manner through the list of the shop offers; the *select* function associated to a *zoom* gesture executed with both hands for enlarging/displaying the details of a particular item; and *back* represented by a simple waving hand gesture for restoring the application current view to the previous state. The same general functions could also apply for the 3rd scenario for which additional specialized gestures could be creatively imagined. For example, Andrew could define a custom posture related to his favorite game character and associate it with that particular game. Once recognized, the system would directly display the latest information about Andrew’s favorite game.

4. A practical implementation for nomadic gestures: The Music Video Shop

In order to demonstrate the concept, we present a free-hand gesture-based interface that allows the prospective clients of a music shop to access, play, and purchase music videos displayed on an ambient screen. The functions that the Music Video Shop exposes are typical for such applications and include:

1. **Navigation.** Browsing the music videos collection available in the store is performed using the *next* and *previous* commands which bring new items into view;
2. **Manipulation.** Videos can be *played* and *stopped*; more information can be requested for the album or the artist currently playing in order to inform the purchase (the *info* command); and, finally, videos can be added to the shopping cart (the *cart* command);
3. **System commands.** When finished, the user logs out the shopping mode using the *log out* command.

Figure 3 illustrates a snapshot of the running application: users browse videos, play and request album information, and finally add videos to the shopping cart. A visual feedback of the user is displayed in form of a body silhouette while the motions performed by the two hands are highlighted using a contrasting color.

4.1. Gesture acquisition, recognition, and uploading

Gestures are acquired using the Kinect sensor of the XBox console [23] which provides color and depth in-
two motion gestures \( G \) and \( Q \) with respect to the sum of individual Euclidean distances between their 6D points:\(^1\)

\[
dtw(G, Q) = c_{|G|-1,|Q|-1}
\]

where \( c \) represents a cost matrix of size \(|G| \times |Q|\) which contains at location \((i, j)\) the optimum (minimum) alignment cost between the subsets \(\{0..i\}\) and \(\{0..j\}\) of the two gestures \( G \) and \( Q \). The values of the cost matrix are iteratively computed using dynamic programming:

\[
c_{0, 0} = \|G_0 - Q_0\|; \quad c_{i, 0} = c_{i-1, 0} + \|G_i - Q_0\| \quad \text{for} \quad i > 0
\]

\[
c_{0, j} = c_{0, j-1} + \|G_0 - Q_j\| \quad \text{for} \quad j > 0
\]

\[
c_{i, j} = \min\{c_{i-1, j}, c_{i-1, j-1}, c_{i, j-1}\} + \|G_i - Q_j\| \quad \text{for} \quad i, j > 0
\]

where \(\|a - b\|\) represents the Euclidean distance between points \( a \) and \( b \) from \( R^6 \).

While this procedure of segmenting gestures excludes many other motions which can be performed in the plane of the body, it is sufficient for demonstrating our concept of nomadic gestures. Note that neither the acquisition or recognition technologies are important here but rather the discussion of the nomadic gestures concept. We therefore limit our discussion on acquisition and recognition aspects here and instead refer the interested reader to excellent surveys of such techniques [18,45].

In the context of our implementation, the user preferred gestures are stored as a file on the mobile device. We used a HP iPAQ 614 Business Navigator for our implementation. A simple application was developed for Windows Mobile 6 in order to upload the file to the ambient display via the TCP/IP protocol using the integrated WLAN 802.11b/g communication of the smart phone and knowing the IP address of the system running the ambient display. Figure 4 shows an illustration of the procedure.

4.2. Matching user gestures and system functions

Once the gestures have been transmitted to the ambient display, a matching procedure must run in order

---

\(^1\)3D for the left and 2D for the right hand.
commands such as help the BASIC-CMDS set but would probably also implement specific application would implement commands from as an initial reference for interactive applications. A not necessarily need to be complete. Instead, it acts, delete and object specific commands such as open and command specific commands (such as add to cart as in our Music Video Shop demo). We denote this set as the BASIC-CMDS set. For example, BASIC-CMDS can contain frequent navigation commands such as previous and next, up and down, and left and right; system commands such as help, log in, log out, and shut down; and object specific commands such as open, cut, copy, paste, delete, etc. The basic set of commands does not necessarily need to be complete. Instead, it acts as an initial reference for interactive applications. A specific application would implement commands from the BASIC-CMDS set but would probably also implement specific commands (such as more info or add to cart as in our Music Video Shop demo). We denote the application set of commands as APP-CMDS. Finally, users define their own gestures for given commands as they go along and as they need a specific command expressly. We denote the set of commands for a specific user as USER-CMDS which initially represents a subset of BASIC-CMDS. Any matching algorithm must compute an optimum association between APP-CMDS and USER-CMDS in order to maximize reuse and minimize the need for defining and training new gestures for new commands. In the following we illustrate the matching procedure in detail by discussing several examples.

As mentioned, the ambient interactive application exposes a set of commands APP-CMDS. For example, the APP-CMDS set of our Music Video Shop includes the following commands:

\[
\text{APP-CMDS} = \begin{cases} 
1. \text{next} & \text{right (.90), up (.25)} \\
2. \text{previous} & \text{left (.90), down (.25)} \\
3. \text{play} & \text{open (.90), maximize (.75)} \\
4. \text{stop} & \text{close (.90), minimize (.75)} \\
5. \text{info} & \text{help (.75)} \\
6. \text{cart} & \text{accept (.50), yes (.25)} \\
7. \text{log out} & \text{close (.50)} 
\end{cases}
\]

for which the meanings have been described at the beginning of this section.

However, it is unlikely that all these commands would find themselves in this form in the users’ own sets but instead synonyms of these commands are more likely to have been defined. For example, some users may already have left and right in their sets which could be very well reused and associated to previous and next as required by APP-CMDS; other users may have already defined help which can be reused for info; and again, others may have defined close which can perfectly serve as log out. All these synonyms must be part of the BASIC-CMDS set. Therefore, in order to maximize reuse of previously defined commands, it makes sense to augment the APP-CMDS with synonyms of equivalent commands that could be used instead of the original ones. Therefore, the augmented set AUGM-APP-CMDS contains APP-CMDS but also contains synonyms from BASIC-CMDS which can be safely used instead of each original command. Figure 5 illustrates the relationships between these sets. For example, the AUGM-APP-CMDS set of the Music Video Shop application may look as follows:

\[
\begin{align*}
1. \text{next} & \quad \text{right (.90), up (.25)} \\
2. \text{previous} & \quad \text{left (.90), down (.25)} \\
3. \text{play} & \quad \text{open (.90), maximize (.75)} \\
4. \text{stop} & \quad \text{close (.90), minimize (.75)} \\
5. \text{info} & \quad \text{help (.75)} \\
6. \text{cart} & \quad \text{accept (.50), yes (.25)} \\
7. \text{log out} & \quad \text{close (.50)}
\end{align*}
\]

for which each synonym also comes with a confidence value in [0.0, 1.0] showing the confidence of the application designer in substituting the original command with the given synonym. In our example, the play command (which starts a music video) will essentially determine the same effect as open or maximize. Also, asking for more info for the current album may be viewed as equivalent with executing a help command. The synonyms list are ordered by their appropriateness for each command as given by their confidence values.
in $[0..1]$. The use of the maximum confidence value of 1.0 is restricted for perfect matchings only.

The matching algorithm must associate user commands (USER-CMDS) with system commands using synonyms to help the process (AUGM-APP-CMDS). The goal of matching is to reuse as many user commands as possible in order to minimize learning and time-consuming training procedures. We can formulate the matching problem in terms used by the graph theory by associating vertices to each command $i \in$ USER-CMDS and $j \in$ AUGM-APP-CMDS and initializing the weight of edge $(i, j)$ with the confidence of matching command $i$ with command $j$ (the rest of the edges are initialized with $-\infty$). The problem consists now in finding the maximum weight matching in the bipartite graph composed of sets USER-CMDS and AUGM-APP-CMDS which can be computed optimally using the Hungarian algorithm as described in Papadimitriou and Steiglitz [41, p. 248] or by any heuristic as in Davis [4].

The matching algorithm will output the best set of associations between user and system commands however it cannot work with associations where they don’t exist (for example, the USER-CMDS set may contain little resemblance to the synonym set). Therefore, the result of the matching will either be a full match or a partial match. We discuss these cases in detail below and comment on the options users have at their disposal by considering practical examples:

(1) **Perfect matching.** Jake is a frequent user of the Music Video Shop and therefore he already defined in the past all the commands exposed by the APP-CMDS set of the application. Whenever he uploads his gesture set, a perfect match is found by the system;

(2) **Perfect synonym-aided matching.** Claude has never used Music Video Shop before however she used other similar ambient displays located in different shops. Therefore, she has already defined next and previous for browsing items as well as cart for adding items to the shopping cart and log out for finishing the purchase. As she never shopped for music videos before, she doesn’t own commands to play or stop the videos. However, she used maximize and minimize before when looking at the various details of fabric when shopping for clothes which allows the system to safely perform the new synonym associations. After uploading the gesture set, Claude is alerted on her mobile phone about the new maximize-play and minimize-stop associations which she confirms with a tap. Claude can immediately start browsing and shopping for music videos. As the synonyms helped identify matchings between commands, we denote this case as a perfect synonym-aided matching;

(3) **Partial sufficient matching.** Susan is also a first-time user of the Music Video Shop however she doesn’t want to purchase any videos but only wants to browse and play. As she does such browsing quite a lot in different shops, she already has defined next and previous for browsing as well as play and stop. As she uploads her gesture set, the system sends back an alert to her mobile showing that no match could be found for cart. However, as she doesn’t want to shop, she decides to ignore the missing associations and starts using the system. We denote this scenario as a partial sufficient matching;

(4) **Partial insufficient matching.** Jim’s gesture set is similar to Susan’s however Jim does want to purchase videos so he needs both info and cart commands. Instead of ignoring the missing associations, Jim checks the missing commands on his mobile phone in order to instruct the ambient display that he needs training to occur prior to system usage. Training consists in acquiring the
same gesture multiple times (see Fig. 6) and storing everything back to Jim’s phone. As Jim had to define new commands and go through training, we denote this scenario as partial insufficient matching.

Figure 7 summarizes the matching process and the different decisions of the user according to the four identified scenarios.

4.3. Video

A video demonstration showing the workings of the Music Video Shop application while describing the Nomadic Gestures concept can be downloaded from http://www.eed.usv.ro/~vatavu/videos/NomadicGestures.flv.

Fig. 6. Training in the Music Video Shop application: (A) and (B) show the user performing info and cart commands; (C) and (D) show the feedback guiding the training process. In this case, the user draws a question mark for info and a check gesture for cart.

5. Informal study

An informal study was arranged and conducted in order to collect observations from potential users with respect to the opportunity and applicability of the concept. A small group of participants was presented with a description of the nomadic gestures concept and asked to comment and contribute on its applicability opportunities. Each participant was asked to think of and state up to three strong points and weaknesses of the concept. Also, participants were encouraged to propose new functionalities as well as to imagine possible problems that could appear during the functioning of such a system. They were also asked to think of possible usage scenarios.

With regards to the strong points of the concept, the participants noted the ease of use in the sense that users don’t need to test or explore the system in order to find out its commands as well as the flexibility of the interaction: users don’t have to learn a new gesture set and the user doesn’t have to know other gestures than those defined by himself. One participant called the process of repeatedly training recognizers as being fastidious. Some participants suggested a reduced interaction time due to the fact that exchanging information to and from the system will take place faster and thus more efficiently while others noted that the flexibility of the interaction... improves. Participants considered the concept as really useful for free-hand gestures and maybe less but still valuable for touch-based gestures.

Fig. 7. The decision process following matching the user’s set of commands USER-CMDS and application set AUGM-APP-CMDS.
that they perceived as being more intuitive (due to the large proliferation of mobile phones and tablets with touch-sensitive interfaces).

With regards to the weaknesses of the concept, one participant argued that users must define their own sets...[and]...possess a new generation mobile phone but he also marked these problems as being minor. Most participants noted the fact that users need to upload themselves the gesture set to the ambient display and that this needs to be done before each interaction. Concerns were also expressed with regards to the shareability of the ambient display in case of multiple users trying to access information at the same time but these relate to somewhat different problems discussed elsewhere in the literature such as in the works of Vogel and Balakrishnan [56], Izadi et al. [17], and Scott et al. [49] on territoriality and sharing the space of interactive displays.

Participants were also asked to think of and propose further improvements for the nomadic gesture concept. The suggestions were related to the above mentioned problems such as to enable automatic transfer of the gesture set and eventually an option for memorizing the user (as a requirement for memorizing the gesture set) for future interactions with the same display: the system could save the gesture set and automatically use it when the person comes again, just like cookies are used in web browsers. One participant commented that it will become trying to perform the same transfer with multiple displays while other said that users could forget or simply not have the time to involve in the process of transferring the gesture set.

The informations gathered in this informal study are useful in order to drive future improvements of the nomadic gesture concept with the most notable one being the automatic transfer of the gesture set to the ambient display.

6. Conclusions

The paper introduced a new concept for reusing gesture commands in unfamiliar environments. The concept can be easily implemented in the form of a reusable set of gestures (e.g. configuration files) that are carried by users on their mobile phones and which are being uploaded to the ambient display prior to interaction. Future work will explore techniques for performing automatic gesture transfer between users and the ambient display at the moment when interaction intent is being detected. Also, new usage scenarios will be explored. The nomadic gesture solution opens up the way to personalized interaction with unfamiliar interfaces and unknown environments by building on the concepts of adaptation and personalization of ambient intelligence.

Acknowledgement

This paper was supported by the project “Progress and development through post-doctoral research and innovation in engineering and applied sciences – PRiDE – Contract no. POSDRU/89/1.5/S/57083”, project co-funded from European Social Fund through Sectorial Operational Program Human Resources 2007–2013.

References


R.-D. Vatavu / Nomadic gestures: A technique for reusing gesture commands


