3D Gesture Recognition: An Evaluation of User and System Performance

Michael Wright, Chun-Jung Lin, Eamonn O'Neill ,
Darren Cosker and Peter Johnson

[1] Department of Computer Science,
University of Bath,
Bath, BA2 7AY, UK
{maew20, mcscjl, eamonn, d.p.cosker}@cs.bath.ac.uk
p.johnson@bath.ac.uk

Abstract. We report a series of empirical studies investigating gesture as an interaction technique in pervasive computing. In our first study, participants generated gestures for given tasks and from these we identified archetypal common gestures. Furthermore, we discovered that many of these user-generated gestures were performed in 3D. We implemented a computer vision based 3D gesture recognition system and applied it in a further study in which participants used the common gestures generated in the first study. We investigated the trade off between system performance and human performance and preferences, deriving design recommendations. We achieved 84% recognition accuracy by our prototype 3D gesture recognition system after tuning it through the use of simple heuristics. The most popular gestures from Study 1 were regarded by participants in Study 2 as best matching the task they represented, and they produced the fewest recall errors.

Keywords: Gestural interaction, 3D gesture recognition

1   Introduction

This paper reports an investigation of gesture as an interaction technique in a pervasive computing environment. We conducted a linked series of empirical studies and system development investigating gestural interaction in a pervasive computing environment. In phase 1 of the research, we sought to identify a candidate set of gestures that could be useful and usable across a range of devices, services and contexts. We asked participants spontaneously to generate gestures to perform given interaction tasks. The tasks were selected through a process of iterative scenario generation and refinement, and ranged from concrete tasks familiar to computer users, e.g. "Select ...", to more abstract tasks, e.g. "Show me a ...". We recorded the gestures made by each participant and categorized typical or most common gestures for the different tasks. In addition, we discovered that many of the gestures were 3-dimensional.

In the next phase, we implemented a computer vision based 3D gesture recognition system and trained it using the set of archetypal gestures derived from the study in phase 1. The system uses 3D cameras to capture a user's hand movements, and Hidden Markov Models (HMMs) to recognize the gestures. Participants were trained on the gestures and then asked to perform interaction tasks using only these gestures.

We collected data on user performance (recalling the correct gesture), user ratings of how well a gesture matched the task being performed and system recognition rates.

Typically there is a balance of cost or effort between the user and the system for a given performance and different approaches tend to put more of the burden on either the user or system while attempting to find an acceptable balance and adequate performance. For example, handwriting recognition systems on mobile devices, such as Graffiti on previous Palm devices, forced the user to form letters in a non-standard way, increasing the burden on the user in order to reduce the burden on the recognition system. Therefore, in the final phase of our study we performed a comparative assessment of the ability of users to remember and perform gestures, the accuracy of the system in recognizing the gestures and the balance achieved between burdening the user and burdening the system for a given level of overall performance and user satisfaction.

2 Background & Motivation

Our research is focused on exploring gesture as an interaction technique for pervasive computing environments. Our work focuses on gestural interactions that range from traditional desktop metaphor interactions (e.g. select, open, move) to more abstract or conversational interactions (e.g. "take a picture of …", "show me information about …").

Categorization of gestures allows us to explore opportunities to exploit the characteristics of different types of gesture for different types of interaction. Kendon [2] describes a set of gesture categories, (gesticulation, language-like gestures, pantomimes, emblems and sign language), which range in their formalism. For example, gesticulation is "free form gesturing which typically accompany verbal discourse" and sign language contains a complete grammatical specification. Other categorizations include those used by Efron [6] and McNeill [7].

These categorizations of gestures allows us to explore the characteristics of gesture such that they can be exploited. For example, [9] examines different categorizations of gesture in order to produce realistic interactions between humans and Artificial Intelligence agents while [13] defines a vocabulary of gestures to be used when interacting with a gesture interface.

Our ultimate aim is to allow users to interact more naturally in pervasive computing environments with more complex interactions. Exploiting the features of these different categorizations may enable these types of interactions. For example, the types of interactions that might be supported range from selecting an image on a large display (manipulative) to taking a photo (iconic), to pointing at a street sign and asking "show me where I am on a map" (gesticulation), to missing an announcement over the public address system in a railway station and cupping your hand behind your ear and pointing at your mobile phone to stream the announcement to the phone (pantomime).

However, the majority of the literature on gesture focuses on the technology used to capture gestures made by the user. Such technology includes accelerometers, infra-red tracking, data gloves, and cameras. The largest body of literature on systems for gesture recognition uses computer vision algorithms with 2D and 3D cameras. For example, [3, 4, 8, 11, 12] describe systems which use HMM models with 2D or 3D cameras in order to capture and recognize gestures made by a user. Wu and Huang [18] provide a review of vision-based gesture recognition systems and techniques.

In each of these systems, the gestures are defined by the designer. As Wobbrock [10] articulates, although these gestures are designed skillfully, they are often designed with priority given to system recognition rates rather than to the users'

requirement for gestures that they feel fit the actions being performed. Palm's Graffiti is another example of this, as is MIT's Sixth Sense [15]. Sixth Sense utilizes vision based gesture recognition techniques to enable the use of gestures to interact with a system. Users can use gestures to perform actions such as taking a photo and controlling user interfaces projected by the device on to different surfaces. These gestures are defined by the system designers and rely on physical and desktop metaphors. This is a reasonable design decision, however as Wobbrock highlights, in a technology which is maturing into commercial systems and products there is a need to explore the gestures that users find most appropriate for given tasks.

Wobbrock's observations expose a gap in the research where gestures are often not designed based on user preference or need but rather on the needs of the system. Although gestures are designed based on a principled design approach, as his study illustrates, even experienced designers cannot predict a gesture set that can fully meet user expectations of interaction.

Similar studies into user defined gesture sets have been undertaken by Fikkert [17] and Kray [5]. Fikkert describes a wizard-of-oz study in which users were asked to perform gestures to control the pan and zoom of a map interface on a large display out of reach of the user. They also conducted a user survey in which participants rated different proposed gestures for 6 different commands when interacting with a large display at a distance. Based on these studies they propose an initial gesture set for interacting at a distance with large displays, based on agreement amongst users both in the generation and in the rating of gestures.

Similarly, Kray describes a study where users were asked to perform gestures using a cell phone to interact with other cell phones, large displays and interactive tabletops. Again, they propose a gesture set based on agreement amongst participants. Further to this study they also assess the ability of cell phones to recognize the gestures in this gesture set and provide design recommendations for sensor hardware to be incorporated into future cell phones.

In all three of these studies it was observed that users produce similar gestures for tasks. Based on this observation, we explored user-generated gestures for interaction in pervasive computing environments. This extends the work done by Wobbrock, Fikkert and Kray by exploring more general interactions in an environment where there are potentially many different devices (e.g. large displays, audio, embedded sensors) and services (e.g. location tracking, travel information etc).

Additionally, we also set out to apply user generated gestures to an implementation of a gesture recognition system and to explore the trade off between the user requirement of natural gestures that fit the action being performed and the system requirement of gestures which can be effectively recognized. This extends the assessment conducted by Kray in that it applies the gestures to a working gesture recognition system and explores the requirements and adaptations needed by both the user and the system.

3   Study 1: Generation of a Common Gesture Set

Prior to running this study, we collaborated with colleagues in several academic and industrial organizations to develop a set of scenarios that explore the ways in which future users interact with pervasive computing. The scenarios focus on the theme of Augmented Travel where multiple devices, services and users come together to enable and enhance the traveling experience from booking tickets to providing contextual information while *en route*. From these scenarios we abstracted example tasks for our study. The tasks included:
- Move a [document/image/advert] from one device to another

- Go back to the previous [page in a document/image/advert]
- Show me the location of this cafe

Study 1 was a generative empirical study in which participants proposed gestures to perform the tasks drawn from the scenarios. Tasks ranged from concrete tasks familiar to computer users, e.g. "Select ...", to more abstract, e.g. "Show me a ...".

Twenty two participants took part in the study, aged from 20 to 44 with a mean age of 29. 16 participants were male and 6 were female. All participants were recruited from around the University of Bath.

Participants were asked to imagine themselves performing the tasks in the course of interacting with a pervasive computing environment. They were asked to visualize the interfaces and objects they might be interacting with. They were deliberately not provided with 'props' or interfaces in order to focus the participants on generating gestures that would allow them to perform the task rather than focusing on the gestures that could be made to interact with a specific interface or object.

Participants were run individually. Each participant was provided with the context in which she should imagine herself performing the gestures. The experimenter read aloud each task in turn and the participant made a gesture of her own choice to perform the task. A subset of the tasks is presented in Table 1. The order of the tasks was randomized for each participant. The gestures performed by each participant were video recorded for later analysis.

Table 1. A subset of the 68 tasks presented to participants in Study 1.

Task No	Task
2	Go to an image
4	Select
17	Zoom in to an image
25	Close
26	Close an application
39	Show me information about this cafe
41	Show me my location
51	Move an application from one device to another
52	Go to an image and zoom in
53	Select a piece of text and delete it
54	Open a document and select a piece of text
57	Zoom in to a map and show me my location

3.1 Results

The resulting video record was analyzed to investigate the gestures generated by participants. In analyzing the gestures, we were particularly interested in the similarity of gestures made across participants for a particular task. Here we focus on the 'verbs' in the tasks, i.e. Select, Move, Go To, as these gestures are the actions or manipulations the participant performed on an imagined interface or object.

Two researchers independently analyzed the resulting video and produced descriptions of the gestures made by participants for the verb in the task. To ensure that the resulting categorization of gestures was based on the same observed gesture we ran an inter-rater reliability test. Each researcher gave a description of the gesture made for each task. These descriptions were then compared and a Kappa statistic was produced to determine consistency between the researchers. The results of the test indicate a very high level of agreement (Kappa = 0.818, $p < 0.001$) between the descriptions of the gestures performed by each participant.

Tables 2 to 4 respectively present the 3 top level categories we identified from our analysis of the gestures. Category A consists of tasks for which a single common

gesture was used by more than 65% of participants for the given task, and the overall variance (i.e. the number of different types of gestures performed) was low. Category B consists of tasks for which the variance was low but there was not a single dominant gesture as there was in Category A. Category C consists of tasks for which the variance was high.

In Category A (Table 2) for each of the actions Select, Open, Close, Stop, Pick Up, Drop and Move, participants typically made one gesture. Furthermore, there is a low variance in the gestures made, i.e. there are few alternative gestures. In all but one case (Open) the variance is 1 if we exclude outliers, i.e. where a gesture was made by only one participant. Thus, for these tasks in the context of the study there was a high level of agreement across participants on the archetypal gesture for this task.

In Category B (Table 3) there is low variance (between 2 and 3) for each of the gestures generated for the tasks Zoom In, Zoom Out, Move Forward, Move Back and Go Back. The cause of variance in this category is primarily due to the direction in which the gesture was performed. For example, both the Zoom In and Zoom Out gestures were performed either as a movement of the hands forwards and inwards to a point or spreading apart outwards from a point. One possible explanation for this variance is the interaction metaphor used by the participant. In the Zoom examples, either gesture could be used depending on the metaphor employed by the participant, e.g. magnifying glass or stretch to zoom. In selecting an archetypal gesture for the Zoom gestures, we added together percentages from the forwards and inwards movement and the outwards and further apart movement and selected as the archetype the higher percentage. Therefore, Zoom In is defined as a movement of the hands forwards and inwards to a point as this direction was used by 48% of participants whereas the movement of the hands spreading apart outwards from a point was 35%. There is no *a priori* reason not to prefer the opposite direction for the Zoom gestures but, in this category, direction is the main distinguishing feature and so the most common direction was used to select the archetypal gesture.

In Category C (Table 4) there is large variance (between 4 and 6) for each of the gestures generated for the tasks Go To, Search, Turn On, Turn Off, Play, Show Me and Delete. The point gesture was performed for almost all of the tasks. One explanation is that actions such as Turn On, Go To etc were, for our participants, considered as equivalent to selecting the object. However, in tasks such as "Show me information about this cafe" the point gesture was used as a default when the participants struggled to think of an appropriate gesture for the task. Hence, it seems more likely that pointing is in many of these cases a symptom of participants' not articulating the specific meaning of the task through the gesture, rather than the various tasks being semantically equivalent to selecting. In determining the archetypal gesture for Category C tasks, we simply chose the gesture generated the greatest number of times by the participants, disregarding the point gesture. These gestures are effectively arbitrary and it is therefore likely that they will be more difficult to learn and remember than Category B gestures where there was less variance, and Category A where there was even less.

Table 2. Category A: Gestures produced in Study 1 for which there is low variance and a greater than 65% concurrence by participants on the gesture for a given task.

Action	Gesture Made	% used
Select	**point**	**86%**
	sideways movement	13%
	circle	1%
Open	**movement outwards like a book**	**71%**
	double tap	9%
	point	12%
	open hand/flash	5%
	upwards movement	3%
Close	**movement inwards like a book**	**73%**
	x shape	22%
	close hand	5%
Stop	**"halt!" sign**	**86%**
	cutting motion	2%
	point	11%
Pick Up	**grasp and pick up**	**80%**
	upwards movement	9%
	sideways movement	11%
Drop	**open hands and a movement down**	**66%**
	push down movement	30%
	x shape	5%
Move	**movement from side to side**	**100%**

Table 3. Category B: Gestures produced in Study 1 for which there is a low variance in the number of gestures produced but there is no single gesture which was generated by participants more than 65% of the time.

Action	Gesture Made	% used
Zoom In	**movement forwards towards a point**	**42%**
	movement inwards like a book	**6%**
	movement from the user outwards	24%
	movement outwards like a book	11%
	pinch	17%
Zoom Out	**movement from the user outwards**	**47%**
	movement outwards like a book	**12%**
	movement forwards towards a point	18%
	movement inwards like a book	9%
	pinch	14%
Move Forward	right to left movement	18%
	left to right movement	**36%**
	z axis forward movement	25%
	circle	14%
	physically move forward	7%
Move Back	**right to left movement**	**36%**
	left to right movement	18%
	z axis backwards movement	25%
	circle	16%
	physically move back	7%
Go Back	left to right movement	11%
	right to left movement	**41%**
	z axis backwards movement	25%
	physically move back	7%
	circle	16%

Table 4. Category C: Gestures produced in Study 1 where there is a large variance. In addition, the point gesture is typically used as a default.

Action	Gesture Made	% used
Go To	**sideways movement**	**36%**
	physically move	11%
	point	41%
	double tap	3%
	icon of object e.g. media or tv	9%
Search	point to eye	6%
	shrug	5%
	question mark (?) icon	17%
	circle	**44%**
	side to side in a z shape	14%
	downwards or sideways movement	15%
Turn On	**turn of the wrist**	**16%**
	up movement	9%
	open hand/ flash	9%
	point	61%
	open gesture	5%
Turn Off	**turn of the wrist**	**20%**
	downward movement	11%
	eyes	5%
	x shape	11%
	two handed large cross movement	7%
	point	32%
	close gesture	14%
Play	point	48%
	open gesture	7%
	wave	5%
	circle	**16%**
	open hand(s)	2%
	right to left movement	2%
	tap	9%
	icon(thumbs up or triangle play)	11%
Show Me	point	47%
	point at eyes	8%
	shrug/hands open gesture	**22%**
	icon of object e.g. media or tv	8%
	circle	7%
	open hand(s)	9%
Delete	**draw an x shape**	**48%**
	right to left movement	9%
	throw	27%
	rip	3%
	close gesture	8%
	downward movement	5%

Participants performed gestures in a variety of directions and orientations depending on how they visualized the interfaces and objects they might interact with in a pervasive computing environment. For example, the Select gesture often had a different direction depending on where the participant imagined the target object to be located and a different orientation of the hand depending on the type of task (figure 1(a) and 1(b)). Another example is the Zoom In and Zoom Out gestures where, although participants made the same gesture in terms of the direction of movement of their hands, the orientation of their hands could either be vertical towards the ground or horizontal in front of them (figure 1(c) and 1(d)). Existing gesture recognition systems typically operate only with 2D gestures, e.g. [11, 12, 15]. Given the

predominance of 3D gestures in the gesture set we derived from Study 1, there would appear to be a need for gesture recognition systems that can recognize gestures in 3D.

(a) Select a TV (b) Select a but- (c) Zoom in to a (d) Zoom in to an
ton   map   image

Fig. 1. Different directions and orientations performed by participants when asked to generate gestures for different tasks.

In the remainder of this paper we present an implementation of a computer vision based 3D gesture recognition system followed by a further study. In this second study we trained participants on the candidate common gesture set derived from the first study and assessed the ability of the users to remember and perform the gestures, the accuracy of the 3D recognition system in recognizing the gestures, and the balance achieved between burdening the user and burdening the system for a given level of overall performance and user satisfaction.

4 Gesture Recognition System

Our 3D gesture recognition system drew on [12]. In [12] they propose a method by which hand movements can be categorized based on a topology of vectors calculated from the movement of the user's hand. We extended this topology to include the third dimension. Furthermore, in [9] only one hand is tracked; in our implementation we are able to track 2 hands and, therefore, to recognize two handed gestures.

Our gesture recognition system is comprised of two main modules: an image processing module and a HMM module. We used a Bumblebee 2 stereo camera (figure 2(a)) to capture the image of the user performing a gesture. From this image the system extracts the x, y and z coordinates of each pixel and uses color detection to locate and track the user's hands. We convert the RGB colour values for each pixel into the Y'UV444 color space to reduce the effects of changes in lighting. To increase performance and object recognition rates, we perform this conversion only if the z value of the pixel is in an active range based on the clustering of detected pixels. If an individual pixel falls outside the z value range of this cluster then it is rejected.

Following identification of the objects, we apply two more filters. The first filter treats as noise detected potential objects whose total number of pixels is not greater than a predefined threshold. The second filter treats the detected object as static if the distance the object has moved between frames is below a predefined threshold.

In the next stage the system calculates a Gesture Sequence for the movement of the user's hands between frames. This sequence is used as input to the HMM model which returns the gesture whose Gesture Sequence best matches the one performed. In order to capture both hands we produce a separate Gesture Sequence for each hand and the HMM is trained using these separate sequences. The outputs of the HMM predictions for the left and right hands are then examined together.
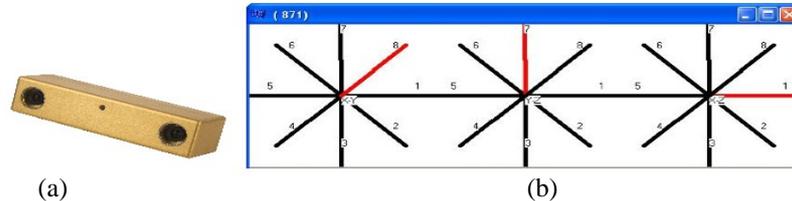
Fig. 2. The system developed used a 3D stereo camera and HMM models in order to capture and recognize gestures in 3D.

In order to encode the hand movements, the system calculates the centre point of the detected object and tracks the movement of this centre point between frames. Using this movement data we divide the x,y,z values into three planes of movement, X-Y, X-Z and Y-Z, with each plane divided into eight directions (figure 2(b)). These coordinates are saved into a buffer. Using this buffer the system is able to calculate the angle between the two movements of an object. Using these angles we can build up a sequence of movements for each axis from one frame to another.

After all the angles have been converted to directions, the system combines the three directions into a number. For example, if x-y is 8, y-z is 7, x-z is 1, the system encodes those directions as 871. Figure 2 shows an example of a section of a Gesture Sequence used as input to the HMM. The HMM module provides a probability that the Gesture Sequence input is a particular gesture. We used the Accord Statistics Library API [1] in order to implement the HMM.

In training mode, the HMM module was given a training set of gesture sequences for each gesture in our candidate set of common gestures derived from Study 1. From the training set the HMM module produces a model for each gesture. In prediction mode, the HMM was given as input the Gesture Sequence derived for each gesture performed by the user. The HMM then identifies and outputs the gesture that has the best match based on the trained gesture sequences.

5 Study Two: User and System Evaluation

Study 2 applied our 3D gesture recognition system to the archetypal gestures derived from Study 1. We aimed to evaluate simultaneously both the participants' performance and experiences in recalling and performing the gestures and the performance of the system in recognizing the participants' gestures. Furthermore, as we report in section 6, we examined the balance between, on one hand, requiring a recognition system effectively to handle the inevitably diverse range of interpretations by users of even a constrained set of gestures and, on the other hand, requiring users to adapt their performance to conform to the equally inevitable constraints of a given recognition system implementation. Historically, some proponents of an 'engineering-oriented' approach have taken the line of 'optimizing' a system's performance at the cost of considerable constraints on allowable user behaviors, while most proponents of a 'human-oriented' approach have argued that the human users should be given more freedom to behave and express themselves as they want, with the system having to cope as best it can. The optimal approach to combining limited machines with diverse humans is probably somewhere between these two extremes.

5.1 Method

Study 2 builds upon Study 1 and uses the gesture recognition system described in Section 4 in order to test the accuracy of the system to recognize the gestures as well as the ability of the users to remember and perform the correct gestures. Participants were trained on a subset of gestures derived from Study 1 (Table 5) and then asked to perform given tasks using only these gestures. As in Study 1 participants were not given any physical devices on which to make a gesture and the study took place in a lab where no devices were present apart from the laptop and stereo camera comprising the gesture recognition system.

18 participants took part in the study, aged from 20 to 44 with a mean age of 30. 14 of the participants were male and 4 were female. All participants were recruited from around the University of Bath.

Participants were run individually. In the first part of Study 2, participants were trained on the set of gestures derived from Study 1 (Table 5). We deliberately removed some gestures from this set so that they could be used in an interference task between training the participants and asking them to complete the tasks.

In the training phase, the participant was asked to perform a specific gesture in front of the gesture recognition system. The participant was shown the gesture by the experimenter and asked to perform each gesture 10 times. Each repetition was recorded by the system and the experimenter made sure that the participant performed the gesture correctly by ensuring the movements made by participants were the same as those demonstrated. In line with our view that human-computer interaction is a 2-way street, it is worth noting that this process trained both the user and the recognition system on the gestures as performed by that particular user.

Following training, the participant performed an interference task in which the experimenter read aloud a task from those previously used in Study 1 (and not otherwise used in study 2), and asked the participant to generate a gesture or gestures they thought corresponded to that task. Participants were encouraged to be as creative as possible in generating these new gestures and they were not constrained to the gestures they had just been shown. Each participant generated gestures for 15 new tasks, taking a minimum of 5 minutes to complete.

Next, the experimenter again read aloud a task, but this time the participant was asked to perform the task using only gestures she had learned in the training phase of Study 2. This was repeated for all the tasks in the training set. The gestures made by the participants were video recorded. The experimenter noted correct gestures made (i.e. that the gestures were recognizable – by the experimenter! – and of the correct type), corrected any mistakes of gesture type (e.g. making a Select gesture rather than an Open gesture) and prompted participants if they could not remember the gesture.

Finally, the participants completed a questionnaire on their experience of the gestures and tasks. In addition, they were asked for their perceptions of how 'natural' they perceived the gestures to be for accomplishing the given tasks.

Table 5. Subset of gestures generated in Study One and carried forward into Study Two.

Gesture	Description
Select	point
Open	movement outwards like a book
Close	movement inwards like a book
Pick Up	grasp and pick up
Drop	open hands and a movement down
Zoom In	movement forwards towards a point
Zoom Out	movement from the user outwards
Move Forward	left to right movement
Move Back	right to left movement
Search	circle
Show Me	shrug/hands open gesture
Delete	draw an x shape

5.2 Results

In keeping with our focus on both sides of the human-computer interaction, we analyzed and compared both the users' performance and the 3D gesture recognition system's performance. We first present results on the system recognition rates and then on the success of the participants in recalling and performing the correct gesture when completing the tasks. Furthermore, we report participants' qualitative preferences in terms of how well they felt the gesture matched the task in each case. In section 6 we compare system and user performance.

System Recognition Rates In order to evaluate the performance of our 3D gesture recognition system we followed a leave-one-out testing strategy to derive an overall accuracy rate as well as a break down of the gestures that were misidentified by the system. Leave-one-out testing involved training the HMM model on all the training data of all but one participant. The omitted participant's data was then input into the system and the output was the identification of the gesture by the HMM based system, from which we were able to evaluate the accuracy of the gesture recognition system.
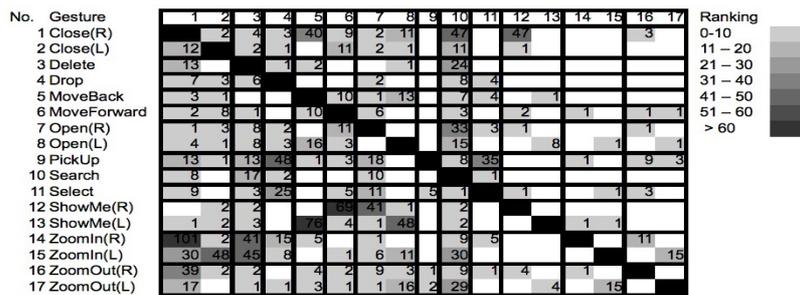
No. Gesture	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1 Close(R)		2	4	3	40	9	2	11		47		47				3	
2 Close(L)	12		2	1		11	2	1		11		1					
3 Delete	13			1	2			1		24							
4 Drop	7	3	6				2			8	4						
5 MoveBack	3	1				10	1	13		7	4			1			
6 MoveForward	2	8	1		10		6			3		2		1		1	
7 Open(R)	1	3	8	2		11				33	3	1				1	
8 Open(L)	4	1	8	3	16	3				15			8		1		1
9 PickUp	13	1	13	48	1	3	18			8	35			1		9	3
10 Search	8	17	2			10							1				
11 Select	9		3	25		5	11		5			1				1	3
12 ShowMe(R)		2	2			68	41	1		2							
13 ShowMe(L)	1	2	3		75		4	1	48			2		1		1	
14 ZoomIn(R)	101	2	41	15	5		1			9	5					11	
15 ZoomIn(L)	30	48	45	8		1	6	11		30							15
16 ZoomOut(R)	39	2	2		4	2	9	3	1	9	1	4		1			
17 ZoomOut(L)	17		1	1	3	1	1	16	2	29			4		15		

Ranking: 0-10 · 11 – 20 · 21 – 30 · 31 – 40 · 41 – 50 · 51 – 60 · > 60

Fig. 3. Confusion Matrix – 61% accuracy: each gesture is shown relative to the gestures with which the system confused it.

Using our initial implementation of the system we achieved an average recognition rate of 61%. Figure 3 shows a confusion matrix for the results of our initial leave-one-out testing. From this matrix we can see that there are a number of cases where the gesture recognition system misidentified a gesture frequently by confusing it with

another similar gesture. By examining the clusters of misidentifications, i.e. where the number of errors is above 30, we can identify some common misidentifications:

1. Zoom In with Close
2. Zoom Out with Close
3. Show Me with Move Back, Move Forward and Open
4. Close with Show Me and Move Back
5. Close with Open
6. Pickup with Drop
7. Pick Up with Select
8. Search in general

To improve the accuracy of the system's gesture identification we applied heuristics to confusions 1-4 in the above list. These heuristics worked because at least one of the gestures being confused was a two-handed gesture. Our first heuristic attempts to correct the confusion between the Zoom In and Zoom Out gestures and the Close gesture. From the confusion matrix we can see that the confusion comes from the misidentification of Zoom In Right Hand with Close Right Hand (101 errors) and Zoom In Left Hand with Close Left Hand (48 errors). However, the reverse is not true, with Close Right Hand and Left Hand not being confused with Zoom In. This is a similar pattern for Zoom Out Right and Left Hand and Close Right and Left Hand. To correct this, we made the Zoom gesture dominant, e.g. if a Zoom In gesture is reported for one of the hands and a Close for the other then the gesture for both hands is assumed to be a Zoom In.

Using the same method as above, our second heuristic made the Show Me gesture dominant over the Move Back, Move Forward and Open gestures. Therefore, if a Show Me was reported for one hand then it was assumed that a Show Me gesture had been performed if the other hand reported either a Show Me, Move Back, Move Forward or Open gesture. Similarly, our third heuristic made the Close gesture dominant over the Show Me and Move Back gestures.

Finding heuristics or improvements in recognition for confusions 5-8 in the above list proved difficult as we could not apply any dominance rules since these gestures are all one handed gestures. The misidentifications in 5 came from each hand being misidentified with its opposite, which we found difficult to correct, e.g. Close Left Hand with Open Right Hand and Close Right Hand with Open Left Hand.

Finally, the Search gesture caused a lot of confusion with all of the gestures. The reason for this is that the Search gesture is a circle. The circle made by the participant, depending on the speed, can mean that the captured Gesture Sequence includes more codes on a particular edge of the circle than on another. For example starting out with the hand at 12 o'clock, rapidly moving it in a circle to 6 o'clock and then slowing down from 6 back to 12 o'clock would produce a Gesture Sequence with more codes that relate to the Gesture Sequence of Zoom Out Left Hand.

Figure 4 shows the results of applying the heuristics in the 3D gesture recognition system. Again we use a confusion matrix to illustrate where misidentification of gestures occurs. The misidentification of gestures is greatly reduced, with the overall accuracy rate increasing from 61% to 84%. As noted in Section 4, we based our system on [9] which had an overall accuracy rate of between 94.29% and 98.6% over a very small set of highly distinct 2D gestures. Our system compares favorably as our 84% accuracy rate was over a larger number and diversity of both one and two handed gestures and in 3D.
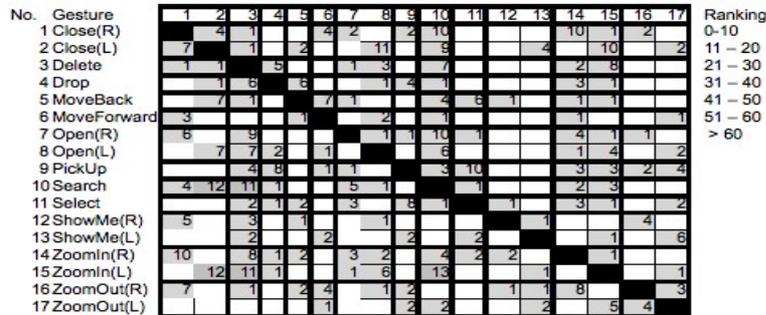
No. Gesture	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	Ranking
1 Close(R)		4	1			4	2		2	10				10	1	2		0-10
2 Close(L)	7		1		2			11		9			4		10		2	11 − 20
3 Delete	1	1			5		1	3						2	8			21 − 30
4 Drop		1	6		6		1	2	1					3	1			31 − 40
5 MoveBack		7	1			7	1		4	6	1			1	1			41 − 50
6 MoveForward	3				1				1					1			1	51 − 60
7 Open(R)	6		9					1	1	10	1			4	1	1		> 60
8 Open(L)		7	7	2		1				6				1	4			
9 PickUp			4	8		1	1			3				3	3	2	4	
10 Search	4	12	11	1		5	1				1			2	3			
11 Select			2	1	2	3	8	1	1					3	1		2	
12 ShowMe(R)	5		3	1			1						1			4		
13 ShowMe(L)			2		2		2		2		2				1		6	
14 ZoomIn(R)	10		8	1	2		3	2		4	2	2			1			
15 ZoomIn(L)		12	11	1		1	6		13		1						1	
16 ZoomOut(R)	7		1		2	4		1	2		1	1	8				3	
17 ZoomOut(L)					1		2	2			2		2		5	4		

Fig. 4. Confusion Matrix – 84% accuracy: each gesture is shown relative to the gestures with which the system confused it.

Participant Data This section presents an analysis of the participants' ability to recall and perform the gestures correctly for the given tasks. In addition, we describe the participants' ranking of how well they thought the gestures matched the actions. Table 6 gives the overall accuracy rate across all participants. It is important to note again that here we are considering correctness of a gesture in terms of whether or not the gesture was of the right type, i.e. a Select gesture when the Select task was intended, rather than whether or not the gesture was recognizable by the 3D gesture recognition system. A gesture made by a participant to perform a particular task could have been of the right or wrong type in these terms. Orthogonally, it might or might not be recognizable as a particular gesture by the recognition system. Thus, the user could intend to perform the gesture for Task A, actually perform the gesture for Task B (poor user performance), and have it recognized by the system as the gesture for Task C or not recognized at all (poor system performance). Thus, accuracy in Table 6 is based on the participants' ability to recall the correct gesture and the experimenter's observation and assessment of the users' performance of the gesture. Table 6 is ordered by incorrectly performed gestures based on the percentage of gestures that participants got wrong.

Category C gestures are clearly mis-performed the largest percentage of times. However, there is a less clear distinction between the mis-performance of Category A and B gestures. The main reason for mis-performing a gesture was the user forgetting the gesture for a given task. This is the main reason for Category C gestures and is not unexpected as these gestures are more abstract than those in Categories A and B. Category B gestures were often mis-performed because participants used the incorrect direction. Again this is not surprising as the cause of variance in Category B was primarily due to the direction in which the gesture was performed.

Surprisingly, since it was a Category A gesture, the Close gesture was often mis-performed. This was often due to the correct gesture being forgotten but in several instances the Delete gesture was performed instead. The Delete gesture was to draw an 'x' shape. A similar shape is extremely commonly used to close a window in traditional desktop user interfaces and it is likely that users' previous experience with this convention overrode their relatively newly acquired gestural metaphor of closing a book. This explanation is corroborated by users' perceptions of how well the gestures matched their associated tasks.

Table 6. Accuracy rate of participants when performing a gesture.

Gesture	Category	% Performed Incorrectly	Reason
Show Me	C	36.11%	Forgot (33.33%), Used select (2.78%)
Search	C	19.44%	Forgot (16.67%), Used move back (2.28%)
Close	A	13.89%	Forgot (11.11%), Used delete (2.78%)
Move Forward	B	11.11%	Used wrong hand (5.55%), Used two hands (2.78%), Used move back (2.78%)
Delete	C	7.41%	Did zoom in (1.85%), Forgot (5.56%)
Pick Up	A	5.56%	Used drop (2.78%), Performed incorrectly (2.78%)
Zoom In	B	3.70%	Used zoom out (3.70%)
Open	A	2.78%	Used Select (1.39%), Included a close gesture (1.38%)
Select	A	2.78%	Dragged over text (0.93%), Forgot (0.93%), Used zoom in (0.92%)
Move Back	B	2.78%	Forgot (2.78%)
Drop	A	0.00%	
Zoom Out	B	0.00%	

Figure 5 shows a ranking of how participants perceived that a gesture matched its corresponding task, ordered by how well the participants rated each gesture. So, for example, 10 participants gave the Select gesture the maximum score of 20, with a cumulative score of 335 for Select. With the exception of Close, participants felt that Category A gestures matched their tasks well. This is as expected since Study 1 found little variance in the user-generated gestures for these tasks. The results of Studies 1 and 2 combined give us some confidence that these are indeed good archetypal gestures for these tasks. There was more variance in the Category B gestures and, again as expected, less agreement on how well these gestures matched their tasks in Study 2. Finally, Category C gestures had the most variance when generated by users in Study 1 and we saw no real consensus amongst the participants in Study 2 that the chosen gestures matched their tasks well. The notable exception here was Delete. Thus, as with performance accuracy, the Close and Delete gestures were the only exceptions to the predicted ranking. The Category A Close gesture was ranked very low while the Category C Delete gesture was ranked high.

Gesture	Category	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	cumulative weighting
Select	A													1		1	1	1	2	2	10	335
Pick Up	A													1	2		3	3	3	6		77
Open	A									1				1	1	1	2	2	5	5		60
Delete	C								1				1			2	4	4	1		5	48
Drop	A															2	3	3	4	1	5	58
Move Forward	B		1				1			1	1					2	3	2	2	1	4	48
Move Back	B		1				1			1	1					3	3	2	1	1	4	42
Zoom In	B			1				3				1	1	1		3	2	1		1	4	34
Zoom Out	B			1				3				1	1	1		3	2	1		1	4	44
Close	A				1									1	1	3		5	4		3	28
Search	C	1		1		2			2	2	2	1		2		1	2	1			1	13
Show Me	C	2	1	3	1	2	1			1	1	1				1	3				1	16

Fig. 5. User ranking of how well the gesture matched the action with 20 being very strong and 1 being very weak.

6 The Trade Off Between System and User

In the previous section we described the results from our second study in terms of both user performance and preferences and system performance. Table 7 presents a comparison between the user ranking of gestures from Study 2 and a ranking of the recognition errors made by the 3D gesture recognition system. A user ranking of 1 represents the best perceived match to the corresponding task and a system error ranking of 1 represents the fewest errors in system recognition of the gesture.

Taken individually, these results could provide design recommendations for the form of the gestures, where the recognition algorithm needs improvement, and even whether gestures should be adopted or rejected. However, the comparison illustrated in Table 7 demonstrates some of the potential conflicts in design recommendations based solely on examining either the user or system performance. For example, the system recognition results would suggest that despite a high user preference the gesture for Pick Up should be changed because of its low system recognition rate. Conversely, the user preference results would suggest that Show Me should be changed based on user ratings that indicate the gesture was not perceived as a good match to the action being performed.

By examining the system and user results together we can begin to explore the potential trade off between the need for gestures that are effective for humans and that are distinct enough to be recognized effectively by a given gesture recognition system. Based on this exploration we can propose a set of design recommendations that take into account this trade off (summarized in Table 8). These recommendations highlight where there is a need to improve the recognition system implementation, alter the characteristics of the gesture (e.g. specifying a particular orientation of the hands) or change the gesture entirely.

Table 7. Comparison of the user ranking of gestures and the misidentification error rate of the system (1 being the highest user ranking and producing the fewest system errors and 12 being the lowest user ranking and producing the most system errors).

User Ranking	Gesture	Gesture	System Error Ranking
1	Select	Move Forward	1
2	Pick Up	Show Me	2
3	Open	Drop	3
4	Delete	Select	4
5	Drop	Delete	5
6	Move Forward	Move Back	6
7	Move Back	Zoom Out	7
8	Zoom In	Open	8
9	Zoom Out	Zoom In	9
10	Close	Close	10
11	Search	Pick Up	11
12	Show Me	Search	12

Table 8. Generalized design recommendations derived from the direct comparison of user and system performance.

System Performance vs User Performance	High	Medium	Low
High	Keep gesture and system in current form	Improve the system and keep gesture the same	Improve the system and keep gesture the same
Medium	Require the user to learn the gesture and keep the system the same	Work could be done on *either* - improving the system performance - tweaking the gesture to allow for better recognition (e.g. orientation of hands)	Work on improving the system, however, if this is not practical or the cost:benefit ratio of doing so is high then the gesture could be altered
Low	Require the user to learn the gesture and keep the system the same	Consider changing the gesture unless there is an easy way of improving the system to recognize the gesture	Change the gesture

In Table 9 we map the results of our study to the general recommendations of Table 8. We then provide an enumerated list of the resulting design recommendations for our 3D gesture recognition system. Recommendations 3, 4, 5 and 6 illustrate the value of considering the trade-off between what works for the user and what works for the system. In each of these cases, simply considering either the users' experience or the system performance alone could have led to very different conclusions.

Table 9. Gestures from our study mapped to the generalized design recommendation table.

System Performance (recognition rate for individual gestures from Study 2) *vs* User Performance (user rating of gesture in Study 2)	High (recognition accuracy between 91-100%)	Medium (recognition accuracy between 81-90%)	Low (recognition accuracy between 71-80%)
High (majority of ratings > 15)	Drop	Select Open Delete	Pick Up
Medium (ratings spread out but most > 15)	Move Forward	Move Back Zoom Out	Zoom In
Low (ratings spread out but most < 15)	Show Me		Close Search

1. **Drop:** This gesture should be retained in its current form as both the user and system performance are high.
2. **Select, Open and Delete:** these gestures are regarded by users as an excellent match to their corresponding tasks. However, the medium system recognition rates indicate that work needs to be undertaken to improve the system.
3. **Pick Up:** similarly, Pick Up should be retained due to its high user rating and work should be undertaken on improving the system.
4. **Move Forward and Show Me:** participants gave these gestures a medium and low rating respectively, indicating that these gestures were only a reasonable or low match to the task being performed. However, both these gestures have high system recognition rates. Therefore, it is recommended that these gestures should be retained and the user should be encouraged to learn the gestures. In the case of Show Me, this is further corroborated by

Study 1 where, setting aside the simple Point gesture as discussed above, the Show Me gesture chosen was easily the most popular gesture generated for this task. Show Me is sufficiently abstract a task that it is unsurprising that Study 2 participants did not rank it highly. It seems likely, again corroborated by the findings of Study 1, that they would have had similar or greater concerns with any other gesture chosen to perform this task.

5. **Move Back and Zoom Out:** the generalized design recommendations suggest that either the gesture or the system could be altered. However, based on the mirrors of these gestures (Move Forward and Zoom In) being retained in their current form, it would seem sensible to recommend that the Move Back and Zoom Out gestures should be retained in their current form and improvements made to the gesture recognition system.

6. **Zoom In:** although the system recognition rate for Zoom In was low, participants reported that the gesture was a reasonable match to the action being performed. Therefore, it is recommended that improvements are made to the system rather than altering the gesture.

7. **Close and Search:** these gestures should be rejected as participants did not regard them as matching their tasks well and the system recognition rate was poor.

7 Conclusions and Future Work

In this paper we have reported a series of empirical studies and system development undertaken to investigate the use of gestures as an interaction technique in pervasive computing environments. In phase 1, participants were asked to generate gestures that we categorized based on the degree of consensus and the number of different gestures generated by participants. Additionally, we discovered that many of the gestures generated by participants were performed in 3D.

Therefore, in phase 2, we implemented a computer vision based 3D gesture recognition system and applied it in a further study in which participants were trained on the archetypal gestures derived from phase 1. Participants were asked to perform tasks using these gestures. From this study we were able to collect data on both user performance and preferences and system performance.

Finally, we explored the trade off between the requirement for gestures to support high system performance versus the requirement for gestures to support high human performance and preference, deriving design recommendations.

Deriving user-generated gestures, as we did in phase 1, enabled us to define an archetypal gesture set for specific types of interactions in pervasive computing environments. The advantage to this approach is that we are able to define gestural interactions that are considered natural and intuitive, based on user expectations and preferences and the degree of consensus amongst participants.

However, considering only the user requirements for gestures when implementing a gesture recognition system for use in pervasive computing environments excludes from the equation the needs of the system. Therefore, we proposed a method by which we could compare both user performance and preference and system performance. The resulting general design recommendations indicate where the archetypal gestures can remain unchanged, where adjustments need to be made to the gesture performance by the user, where development effort is needed to improve a recognition implementation and where a potential gesture could be rejected. We illustrated the application of these general recommendations to our particular gesture set and system implementation.

As part of our future work we wish to define a framework that designers can employ to add new gestural interactions to our archetypal gesture set for new tasks. This framework should not only take into account how to generate gestures for particular tasks but also the practicalities of gesture recognition and interaction. For example, the technology used to recognize gestures (e.g. computer vision with 2D or 3D cameras, accelerometers etc) and the context of the interaction.

Furthermore, we plan to identify further gestures using this framework and evaluate them with a range of gesture recognition systems for pervasive computing environments. The aim is to compare these different systems, exploring the trade off between user and system performance. From these studies, we aim to provide insights into the types of gestural interactions that work well – and poorly – for different recognition technologies in different contexts.

References

1. Accord_Statistics_Library, http://www.crsouza.com
2. A. Kendon.: Current Issues in the Study of Gesture. In: The Biological Foundations of Gestures: Motor and Semiotic Aspects, pp 23-47, Lawrence Erlbaum. (1986)
3. A. Ramamoorthy, N. Vaswani, S. Chaudhury, and S. Banerjee.: Recognition of Dynamic Hand Gestures. In: Pattern Recognition 36(9), pp 2069-2081. (2003)
4. C. Keskin, A. Erkan, and L. Akarun.: Real Time Hand Tracking and 3D Gesture Recognition for Interactive Interfaces using HMM. In: ICANN/ICONIPP 2003, pp 26-29. (2003)
5. C. Kray, D. Nesbitt, J. Dawson and M. Rohs.: User-Defined Gestures for Connecting Mobile Phones, Public Displays and Tabletops. In: MobileHCI 2010, pp 239-248. (2010)
6. D. Efron.: Gesture and Environment. Morningside Heights, New York: King's Crown Press. (1941)
7. D. McNeill.: Hand and Mind: What Gestures Reveal about Thought. University of Chicago Press. (1992)
8. F. Chen, C. Fu, and C. Huang.: Hand Gesture Recognition Using a Real-Time Tracking Method and Hidden Markov Models. In: Image and Vision Computing 21(8), pp 745-758. (2003)
9. I. Poggi.: From a Typology of Gestures to a Procedure for Gesture Production. In: International Gesture Workshop 2002, pp 158-168. (2002)
10. J. O. Wobbrock, M. R. Morris, and A. D. Wilson.: User-Defined Gestures for Surface Computing. In: CHI 2009, pp 1083-1092. (2009)
11. K. Oka, Y. Sato, and H. Koike.: Real-Time Fingertip Tracking and Gesture Recognition. In: IEEE Computer Graphics and Applications, pp 64–71. (2002)
12. M. Elmezain, A. Al-Hamadi, J. Appenrodt, and B. Michaelis.: A Hidden Markov Model-Based Isolated and Meaningful Hand Gesture Recognition. In: Electrical, Computer, and Systems Engineering 3(3), pp 156-163. (2009)
13. M. Nielsen, M. Störring, T.B. Moeslund, and E. Granum.: A Procedure for Developing Intuitive and Ergonomic Gesture Interfaces for HCI. In: International Gesture Workshop, pp 409-420. (2004)
14. M. Wu, and R. Balakrishnan.: Multi-Finger and Whole Hand Gestural Interaction Techniques for Multi-User Tabletop Displays. In: UIST 2003, pp 193-202. (2003)
15. P. Mistry, P. Maes, and L. Chang.: WUW - Wear ur World - A Wearable Gestural Interface. In CHI 2009, pp 4111-4116. (2009)
16. S. Malik, A. Ranjan and R. Balakrishnan. Interacting with Large Displays from a Distance with Vision-Tracked Multi-Finger Gestural Input. In: UIST 2005, pp 43-52. (2005)
17. W. Fikkert, P. van der Vet, G. van der Veer and A. Nijholt.: Gestures for Large Display Control. In: Gesture in Embodied Communication and Human-Computer Interaction, pp 245-256, Springer. (2010)
18. Y. Wu and T. Huang.: Vision-Based Gesture Recognition: A Review. In: Gesture-Based Communication in Human-Computer Interaction, pp 103–115, Springer. (1999)