Speech Displaces the Graphical Crowd

Mohammad M. Alsuraihi
Department of Computing, University of Bradford, Bradford, UK
Email: mmnalsur@bradford.ac.uk

Dimitris I. Rigas
Department of Computing, University of Bradford, Bradford, UK
Email: d.rigas@bradford.ac.uk

Abstract—Developers of visual Interface Design Environments (IDEs), like Microsoft Visual Studio and Java NetBeans, are competing in producing pretty crowded graphical interfaces in order to facilitate completion of interface-design tasks. Previous studies have shown that such interfaces cause the user to experience information overload as well as they create a fertile environment for usability problems. In this paper, we empirically investigate speech as an input and output means for solving usability problems with GUI interaction metaphors and enhancing usability of visual IDEs. The empirical investigation aimed at measuring usability of two experimental ID toolkits: typical visual-only (TVOID) and multimodal (MMID) in terms of efficiency, effectiveness and satisfaction. Usability of these two environments was tested twice, over two experimental phases, by four independent groups of users. The first experimental phase was for measuring usability by novel users in order to explore how learnable each of the two experimental toolkits would be. The second phase aimed at measuring usability by well-trained users to study experienced user performance when using a speech recognition ID.

Index Terms—experienced user performance, novel user performance, speech recognition, usability.

I. INTRODUCTION

The user-interface of visual Interface Design Environments (IDEs) like Microsoft Visual Studio .NET are considered to be one of the most crowded interfaces with graphical widgets and visual alerts. This visual crowd causes the user to experience information overload [6, 21] by which important information in distant locations around the interface may be missed [25]. More seriously, a visually crowded interface affects user performance negatively [9, 39] as it creates a fertile environment for usability problems. These include slipping-off over graphical buttons, miss-selection and hitting unwanted menu-items. Usability problems take place when closure\(^1\) happens [7, 15, 16] by which interface intrusion into task\(^2\) is an inevitable result [8, 23, 24].

Previous research on visual solutions for increasing usability of graphical interfaces suggest using ad-hoc graphical metaphors to facilitate learning on how to use the interface [3, 4, 14, 40]. However, these solutions are still visual and add more load on visual channel. Typically, these solutions target novel users and often neglect experienced users who might wish to increase performance. Other modalities that enable the user to interact using other senses are highly demanded. There is a large body of research on multimodality for enhancing user-interface interaction.

Many studies recommend using earcons [31, 32, 37] and auditory icons [17-19, 42] as auditory solutions to enhance usability of graphical metaphors. However, in order for these interaction metaphors to be properly perceived and correctly mapped to the meanings they are designed to convey, the user is required to maintain a high level of concentration and try to develop a correct perceptual context [35, 36]. Therefore, to overcome this problem and hence, increase usability, synthesised speech should be combined with non-speech interaction metaphors [33, 34, 36, 41]. The potential of speech as a natural communication mean in human-computer interaction should not be ‘overlooked’, as it may effectively complement other modalities, like windows, icons, and menus [22]. Vocal instruction can be very advantageous in systems where ‘hands’ and ‘eyes’ are busy all the time [11]. Visual IDEs are instances of such systems that imply full involvement of hand in mouse-actions and eye-focus when designing a form.

HCI researchers have always worried about efficiency and effectiveness of vocal instruction because of speech-recognition problems. As a consequence of this concern, most of the previous studies suggested reinforcing speech-recognition with another modality, pointing or gesture. The work by Bolt in 1980 [5] pioneered the field of enhancing human-computer interaction with speech and pointing. Since then, many studies in the field used this approach to reduce the impact of speech-recognition errors and increase usability. For instance, the QuickSet system [12, 13] was deployed in a tent in a military-field where it was subject to much noise coming from explosions, low-flying jet aircrafts, generators, etc. According to the authors, this caused spoken interaction with QuickSet to be infeasible. Consequently, they needed to involve gesture with speech-recognition.

\(^1\) Closure occurs when the user perceives that an action has been activated when it has not.
\(^2\) Interface intrusion into task happens when the user is forced to move his/her visual attention from what he/she is doing to look for a piece of information.
Several studies by Oviatt et al [26-30] have argued that a speech-recognition interface combining speech with another input modality, namely, gesture would be more usable than using speech only. This is because the frequency of errors caused by speech-recognition problems would then be reduced. Nevertheless, none of these studies have reinforced this finding with a usability testing study. Usability does not only imply measuring efficiency in terms of frequency of errors, but also in terms of task-accomplishment time. It also implies measuring effectiveness in terms of percentage of functions learned and tasks completed successfully; and finally it implies measuring user-satisfaction with the provided interaction metaphors. An empirical study on a speech-and-gesture system presented a comparison between the percentages calculated for using speech-only, pointing and using both modalities during completion of the required tasks [10]. From this study, it was found that most of user-input was in using speech-only 61%, followed by 36% using gesture-only and only 3% of user-input was by using both modalities. Other studies, have shown an indication that speech-recognition errors may not correlate negatively to interface usability and user-performance [1, 20, 38]. Another study has shown that speech-recognition errors could be solved by increasing the level of confidence in voice when uttering words [2]. Furthermore, in our opinion, using gesture technology for input is more expensive and less handy to use by ordinary PC users than using the mouse. Henry Ford once asserted “technology is a real progress when it is available to anyone”.

This paper aims at finding out how usable a speech-only system (IDE) would be if it was combined with text-to-speech as an output modality along with limited use of the mouse, the cheapest and handiest input technology. The paper also investigates the possibility of substituting most of the existing common GUI metaphors, like menus, toolbar, toolbox, properties-table and status-bar with speech. Furthermore, the paper presents an empirical multi group usability study took place over three experimental phases in order to satisfy the research aims. The first experimental phase was aimed at measuring learnability (or the ability to complete tasks from first time use) of two experimental IDEs: visual-only IDE called TVOID and multimodal (visual and auditory) IDE called MMID. The second experimental phase focused on measuring experienced user performance (EUP) of TVOID and MMID. Finally, the third experimental phase concentrated on measuring satisfaction with the visual (graphical) and auditory interaction metaphors offered in the two experimental toolkits.

II. EXPERIMENTAL TOOLKITS

Two experimental interface design toolkits, TVOID (Typical Visual-Only Interface Design) and MMID (Multi-Modal Interface Design), were developed using Microsoft Visual C#. TVOID imitated the style of interaction implemented in most of the existing interface-design environments like Microsoft Visual C# and Java NetBeans IDE. It interacted with the user visually-only with no involvement of other senses like the auditory system. This interaction took place in six areas in its main interface: menus, toolbar, toolbox, workplace, properties-table, and status-bar. MMID was used as a control environment to which usability of MMID was compared. Figure 1 shows a screenshot of TVOID.

Figure 1: A screenshot of TVOID’s main interface

MMID is a speech-recognition and text-to-speech based environment that allows limited use of the mouse and the keyboard. The aim of this toolkit was to explore whether or not the user would become familiar with an IDE that provides with the least possible graphical metaphors, and whether he/she would use it. MMID allowed the user to interact with it from position of the mouse-cursor such that there was no need for the user to use any of the interaction areas provided in TVOID but the workplace area where design-tasks take place. The system command receptor in this environment was represented by a friendly character (MS Agent) that listened to commands and reacted with speech and facial expressions. Vocal commands and feedback spoken messages were in the form of simple one to three English-words. Figure 2 shows a screenshot of MMID.

Figure 2: A screenshot of MMID’s main interface

III. EXPERIMENTAL PROCEDURE

According to ISO standard 9241 [15], usability implies effectiveness and efficiency of, and satisfaction with the
product. These three criteria were tested empirically using both experimental toolkits (TVOID and MMID) during three phases. Efficiency was measured by timing task accomplishment and calculating the number of errors. Effectiveness was measured by calculating percentage of tasks completed successfully within task-criterion times by all users and percentage of users completed all tasks successfully (i.e. within threshold times).

**Experimental Phase I**

The first experimental phase aimed at empirically measuring usability (efficiency and effectiveness) of the toolkits under the condition of first time use. The toolkits were tested independently by two groups of novel users (A & B). Each group consisted of 15 users. The participants were computer users with limited experience in using interface design environments.

Before commencing the experiments, each user attended a 10-minute tutorial video about the system he/she was assigned to test before starting testing it empirically. The users were aware that they had to pay good attention for the tutorial video in order for them to be able to complete the tasks. This training was necessary to teach the users about the unfamiliar environments they were about to use, and to explore how learnable each environment would be in terms of percentage of functions learned in absence of additional help during task accomplishment. After a training session, each user of TVOID or MMID was asked to write down his/her expectations of how the required tasks would be done. The users, in this stage, were not allowed to see the toolkits' interfaces. The purpose of this procedure was to find out how memorable the graphical and vocal interaction metaphors used after learning about them from the tutorial videos.

Each user was then asked to complete 10 tasks. The tasks increased in complexity and covered all expected actions when using an IDE (i.e. activating menu-commands for filing and editing, selecting tools, drawing objects, and setting properties). Data was collected through observation of function learning time, task completion time and number of errors made during accomplishment of each task.

**Experimental Phase II**

The second experimental phase aimed at measuring efficiency and effectiveness of the two experimental toolkits under the condition of experienced use this time. The toolkits were tested independently by two different groups of users (C & D). Each group consisted of 40 users. In this experimental phase, the sample (number of users) in each group was dramatically increased to ensure that it would be more representative of the target population.

The participants were computer users with good experience in using interface design environments like MS Visual Basic, Visual C++ and Java NetBeans. Furthermore, each group of users was well-trained on using their system before commencing the experiments. During experiments, each user was asked to complete 10 tasks increasing in complexity. Data was collected through video-observation.

**Experimental Phase III**

In this experimental phase we aimed at measuring satisfaction with the interaction metaphors provided in the two experimental toolkits. In order to fulfill this aim, these interaction metaphors were assessed empirically by one group consisting of 80 users. Each user had the chance to use the graphical and auditory interaction metaphors used for activating menu-command functionality, drawing, setting properties, and searching for help. Then, a questionnaire was used for taking users evaluation feedback.

**IV. DISCUSSION OF RESULTS**

**Results from Experimental Phase I**

Figure 3 shows the number of tasks their procedure was successfully expected by the users before actually doing them using TVOID and MMID. 33% of TVOID users expected how to do the tasks right. On the other hand, more than half (55%) of their counterparts who used MMID gave the right expectations. This result was found very significant \( t = 3.43, CV = 2.05, P = 0.002 \). The users of MMID were aware, by the tutorial video, that the tasks in this system could be done using simple one-to-three English phrases along with limited use of the mouse. This design made it easier for them to remember and expect the right vocal commands, and when to use them and when to use the mouse. On the other hand, the users of TVOID were aware that this system was offering several graphical interaction metaphors for doing the required tasks using six areas within the interface (menus, toolbar, toolbox, workplace, properties-table and status-bar). This design made it difficult and hesitating for them to expect locations of the right tools and how to access them, recalling that they had very limited experience with similar systems.

Although the users of TVOID were not familiar with the layout of IDE interfaces, nevertheless, they were used to some of the graphical metaphors offered in the system like the menus and the toolbar. They were used to Microsoft Windows basic applications like MS Words and MS Paint. This experience made the users, who did not remember how to do the tasks from the tutorial video, try to expect how to do them based on the experience they had with other graphical systems. Expectations were incorrect sometimes, which caused the users to explore how to do these functions either by trial or by using the help. In this way, the users of TVOID did two things before learning about functions using the system: (1) recalling the training material and (2) expecting how to do the required functions based on their experience with graphical interaction. This was not the case with the users of MMID who did not remember the right vocal commands from the tutorial video. These users were not familiar to voice-instruction; therefore, they headed directly to exploring.
Figure 4 shows that the users of MMID took 0.470 to 2.260 seconds to learn the vocal commands required for activating menu-command functionality. On the other hand, the users of TVOID took more time ranging from 0.760 to 11.64 seconds to access the same commands using graphical menus and toolbar. Gathering vocal-commands in one position (one alphabetically sorted list) within the interface of MMID helped the users to learn the right vocal commands faster than the users of TVOID, who looked for the required commands among hidden menu-items and mysterious iconised toolbar-buttons. This design implied more mouse movements to distant locations from the workplace area.

Figure 5 also shows the average time calculated for finding a tool in order to draw a particular object using the two environments. In TVOID the users had to do two actions in order to learn how to draw an object: (1) find the tool in order to select it, and then (2) find out how to draw it using the mouse by trial and error. In MMID, the users did not have to use the toolbox and then the mouse to draw an object. Most of the users were aware by the training video that they only needed to say one word which is the tool's name in order to draw the required object. Only four of them forgot this information and used the list of alphabetically sorted tool-selection vocal commands. Thus, learning how to select a tool and then draw the object using TVOID took more time (9.66 seconds average time) than learning how to draw the same object using MMID (5.38 seconds average time gathered from the users who used the tool-selection vocal-commands list).

Figure 6 shows the average times calculated for finding the required properties using TVOID and MMID. In TVOID, the user had to read the properties until finding the required one, which implied scrolling for far properties in the properties-table. On the other hand, the users of MMID could find the required vocal command for setting a particular property by saying "property" followed by the property name. The system would then speak the required vocal command. This design helped the users of MMID to access the required properties faster than their counterparts in TVOID as can be seen in Figure 6.

Figure 7 shows the mean values of time observations taken for accomplishing the 10 required tasks from first time use, comprising the time taken to learn functions and the time taken to complete them. The figure shows significant difference (t = 2.64, CV = 2.10, P = 0.02) in task-accomplishment time between the two environments. This difference was a result of the way used for learning functions and accessing

© 2008 ACADEMY PUBLISHER
tools/commands using the two experimental toolkits. MMID kept the user in the workplace and allowed him/her to do all tasks from the current position of the mouse-cursor by enabling him/her to command it by voice and hear feedback and functionality confirmation while concentrating on design-tasks. On the other hand, TVOID forced the user to move the mouse to different and distant locations within the interface to complete tasks as well as it put the whole load on the visual channel during designing objects and during looking for hidden menu-items, tools and properties while designing the interface.

Efficiency was also handled from the angle of errors made in both environments. It was found that using vocal instruction to complete tasks was more error prone than using the graphical metaphors. The users of MMID did 41 errors while the users of TVOID did only 16 errors, and the difference is significant ($t = 2.17, P = 0.04$). All the errors made during using MMID happened because of speech recognition problems which took place because of sensitivity toward noise (other voices around) and accurate pronunciation. One of the factors that cause command misrecognition is the lack of confidence in user’s voice when saying commands. The user worries that the system is not going to recognize what he/she says and thus his/her voice sounds with less confidence. It was noticed that most of the users in Group B had experienced this problem. In spite of this result, it was noticed that making errors during using MMID made the users more used and familiar to voice-instruction as the system was directing them to utter as should be. This factor caused effectiveness of learning functions and, hence, task-accomplishment to increase. Figure 8 illustrates effectiveness of the two experimental environments in terms of percentages of functions learned in absence of additional help and tasks completed successfully.

On the other hand, the users of TVOID (Group A) did most of their errors because of the usability problem closure. 60% of the users were trapped by a tricky dialogue box (Figure 9). This dialogue box showed many check-boxes to be set. One of them was a must checked...
in the second experimental phase. This hypothesis was investigated during task-completion. This hypothesis was investigated in the second experimental phase.

Results from Experimental Phase II

The results obtained from testing efficiency by novel users during the first experimental phase showed that using vocal instruction for designing interfaces was more efficient than using the graphical metaphors in terms of time taken to finish the jobs, but not in terms of error-frequency. The second set of experiments which was done with experienced users has also shown that MMID was more efficient in terms of task-completion time. This can be seen from Figure 10. The difference was found extremely significant \( t(10) = 2.96, CV = 2.10, P = 0.008 \).

![Mean Value of Task Accomplishment Time](image)

Figure 10: Mean values of time taken to accomplish 10 tasks by 80 experienced users using TVOID (Group A) and MMID (Group B)

Additional drawing tasks were required to be finished in this set of experiments to manifest the difference between using the mouse and the voice when drawing. This difference was found noticeable in regard to access time especially when selecting far tools from a currently open toolbox-page and when looking for hidden tools in other toolbox-pages. Selecting tools by voice directly with no need to do a single mouse move for it has caused this difference. Figure 10 also shows significant difference between the average times taken to reach the same required help content in both environments using a typical textual help tool (in TVOID) and a multimodal help tool for searching by speech (in MMID). The number of page-transactions done until reaching the right help-page during using the textual help tool was ranging from 3 to 17 transactions, while only one page-transaction was done to reach the right help-page when using the multimodal help tool by saying the required keyword.

The rate of speech recognition errors has been reduced in this set of experiments because of doing the experiments under the condition of Experienced Use. The users of MMID did fewer errors this time (TVOID: 34 errors & MMID: 25 errors). Based on our experimental observations, it could be concluded that using a number of graphical interaction metaphors frequently until the user is very used to them would increase the probability of error-occurrence more than using them rarely (i.e. by novel users). In our opinion, an experienced user becomes more and more careless when using the same graphical metaphors time after time due to cumulative familiarity. In other words, the user deals with the familiar graphical metaphors with haste because he/she is very used to them, which sometimes causes a usability problem like closure to occur. On contrary, the frequent use of MMID helped in decreasing the rate of speech-recognition errors as it made the users learn how the vocal commands should sound when saying them.

<table>
<thead>
<tr>
<th>Usability Problems</th>
<th>TVOID</th>
<th>MMID</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>n</strong></td>
<td><strong>%</strong></td>
<td><strong>n</strong></td>
</tr>
<tr>
<td>Mouse slipped and unwanted menu-item was hit.</td>
<td>10</td>
<td>29%</td>
</tr>
<tr>
<td>Mouse slipped and unwanted button was pressed.</td>
<td>2</td>
<td>6%</td>
</tr>
<tr>
<td>Mouse released by mistake during drawing and too small control was drawn.</td>
<td>4</td>
<td>12%</td>
</tr>
<tr>
<td>Confusing a property’s value with another of another property.</td>
<td>2</td>
<td>6%</td>
</tr>
<tr>
<td>Data-entry errors</td>
<td>16</td>
<td>47%</td>
</tr>
<tr>
<td>Missing or passing required tool/property during using scrollable-tags.</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>System could not recognize spoken command. None action happened.</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td>System misunderstood spoken command and unwanted action happened.</td>
<td>0</td>
<td>0%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>34</td>
<td>100%</td>
</tr>
</tbody>
</table>

On the other hand, the vocal data-entry in MMID decreased the rate of data-entry errors because saying phrases (vocal commands) did not enquire certain skills or expertise. In addition, vocal data entry was found more
efficient as it saved the time spent in recovering from typing errors and the time consumed in slow typing when using the keyboard. TABLE 1 shows that the probability for occurrence of speech recognition problems (unrecognizing commands and misrecognizing or miss-mapping them correctly) is less than the probability for occurrence of the errors that happen due to using the mouse and the keyboard. Also, it was noticed that the speech recognition errors were strongly related to the user’s ability of pronouncing words and the level of confidence in voice when uttering words.

Table 1: Probability of Speech Recognition Problems

<table>
<thead>
<tr>
<th>Speech Recognition Errors</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recognition</td>
<td>0.05</td>
</tr>
<tr>
<td>Misrecognition</td>
<td>0.03</td>
</tr>
<tr>
<td>Miss-mapping</td>
<td>0.02</td>
</tr>
</tbody>
</table>

Figure 11 illustrates effectiveness of the two experimental environments in terms of percentage of tasks completed successfully (i.e. within task-threshold time) by all users. From Figure 11, it can be seen that 70% of the tasks were not completed successfully during using TVOID. This percentage was mainly obtained because of the delays in recovering from typing errors and the usability problems listed in TABLE 1, and because of delays when moving the mouse between the workspace area and the other areas within the interface to activate the right graphical widgets. On the other hand, all the required tasks were completed within their critical times during using MMID. There were three factors behind this result: (1) testing the environment by well-trained users helped them to become used to the way the vocal commands should sound as well as it helped them utter words with an acceptable level of confidence, (2) MMID limited the use of the mouse and replaced the graphical widgets with vocal commands, (3) it allowed the user to do all jobs within the workspace area with no need to leave it to any other areas within the interface.

Results from Experimental Phase III

Results from analyzing the evaluation given in the form of satisfaction ratings (Very Satisfactory, Satisfactory, Dissatisfactory and Very Dissatisfactory) showed that 73% of the users considered activating menu-command functionality by voice as Very Satisfactory, and 19% of them considered it as Satisfactory. On the other hand, 9% of the users did not appreciate the use of voice-instruction for doing design tasks and considered it Very Dissatisfactory. They rather preferred to use the more familiar interaction metaphors: graphical menus and toolbar. The toolbar was chosen as the second satisfactory interaction metaphor as 60% of the users liked to use it (considered it Satisfactory) and 30% considered it Very Satisfactory. Using the graphical pull-down menus came in third position of satisfaction as 45% of the users considered it Satisfactory and 38% of them considered it as Very Satisfactory. After giving the satisfaction ratings, the users were asked to only choose one interaction metaphor as the most preferred for activating menu-command functionality. Figure 12 shows the percentages of users who chose each of these interaction metaphors as the most preferred. It reinforces the previous results from satisfaction ratings as it shows the same order given for satisfaction.

Figure 12: Percentages of users who preferred to use the menus, toolbar, hot-keys, or vocal-commands as the best interaction metaphor for activating menu-command functionality

Drawing implies selecting a tool, positioning the mouse on a certain location to draw an object and then drawing that object. According to results from the given satisfaction ratings, most of the users (78%) liked to select tools by voice and considered it Very Satisfactory, and 15% of them considered it as Satisfactory. They liked the idea of saying the tool’s name in order to select it and draw it at the same time. In MMID, the user could put the mouse on the location he/she wants and then say the tool’s name to be drawn on that location. This procedure was smoother and more exciting than looking for the required tool visually using the toolbox in TVOID. The typical well-known toolbox was placed, by the users, in the second position of satisfaction (48%: Very Satisfactory & 35%: Satisfactory). However, these results also indicate that most of the users (83%, which is 48% for VS + 35% for S) would like to still use the toolbox for selecting tools, and this returns to the level of familiarity they had with this interaction metaphor. Nevertheless, Figure 13 shows that majority of the users preferred vocal tool-selection more than using the typical graphical toolbox.

Figure 13: Vocal Commands (68%), Tools (35%), Menus (18%), and Hotkeys (15%)
hand, the mouse-coordinates could be specified orally in MMID such that the user could say the command Mouse followed by the required X and Y coordinates and the mouse-pointer would move to the new specified location. Results show that 75% of the users liked this feature and regarded it as Very Satisfactory, and 19% gave it the rating Satisfactory. Using the interactive mouse-cursor was chosen as the second satisfactory tool for specifying mouse-coordinates. Almost half the sample (49%) rated it as Very Satisfactory, and 38% gave it the rating Satisfactory. Using the status-bar was the least appreciated as only quarter of the sample (25%) considered it Very Satisfactory, and 38% considered it Satisfactory. 41% of the users did not like to use this interaction metaphor for specifying drawing location. Figure 14 shows that most of the users (60%) chose vocal location specification as the most preferred interaction metaphor for this purpose.

![Vocal Commands, 65%](image1)

**Figure 13:** Percentages of users who preferred to use the toolbox or vocal-commands as the best interaction metaphor for selecting tools when drawing

![Reading mouse-coordinates from status-bar, 6%](image2)

**Figure 14:** Percentages of users who preferred to use the status-bar, mouse-cursor, or vocal-mouse-coordinates as the best interaction metaphor for specifying drawing location

Drawing in TVOID can be done as in similar existing systems by using the mouse (click once or press-drag-and-release), while in MMID it can be done vocally by saying the name of the required tool. Less than half of the users (43%) rated using the mouse for drawing as Satisfactory and 45% of them considered it Very Satisfactory. On the other hand, most of the users (74%) liked to draw by voice and considered it Very Satisfactory, and 19% of them gave it the rating Satisfactory. Figure 15 demonstrates this result in terms of the most preferred mean for drawing.

After assessing the drawing interaction metaphors, the users assessed the interaction metaphors used for setting properties for drawn objects (properties-table using TVOID and vocal properties using MMID). According to results, less than half of the users (45%) considered using the properties-table as Very Satisfactory, and almost the same percentage (46%) considered it Satisfactory. On the other hand, 68% of the users described setting properties by speech as Very Satisfactory, and 23% thought it was Satisfactory. Figure 16 illustrates the percentages calculated for user-preference given for both interaction metaphors, which shows that vocal property-setting was chosen as a more preferred way for setting properties than using the typical graphical properties-table.

![Toolbox, 35%](image3)

**Figure 15:** Percentages of users who preferred to use the mouse or vocal-commands as the best interaction metaphor for drawing

![Properties-table, 33%](image4)

**Figure 16:** Percentages of users who preferred to use the properties-table or vocal-commands as the best interaction metaphor for setting properties

Using help was assessed from two angles: (1) the way the user searches for particular help-content using a keyword, and (2) the way the system conveys the required help content. In these experiments, searching for help could be done textually using TVOID and auditorily using MMID. On the other hand, the required help-content was conveyed to the user visually through text and static images in TVOID, and visually and auditorily using interactive self-training multimedia (interactive video) in MMID. According to results, 68% of the users considered searching by voice as Very Satisfactory and 29% of them gave it the rating Satisfactory. Only 3% of the users (2 out of 80 users) disliked this feature and preferred to search by text as usual. On the other hand, 67% of the users also liked to still using the typical textual help for searching for keywords (23%: VS and 44%: S). However, answers collected for the question about the most preferred mean for searching for help,
show that most of the users (80%) preferred to search for keywords by voice rather than to search for them textually, as can be seen from Figure 17.A.

Results also show that all the users liked the interactive multimedia help, and the majority of them (78%) considered it Very Satisfactory. On the other hand, less percentages was calculated for the users who liked reading textual help-pages (26%: VS & 45%: S). Figure 17.B reinforces these results and shows that almost all of the users (93%) preferred to learn using interactive multimedia more than learning by reading text.

[Graph showing percentages of users who preferred to search for help textually or vocally, and who preferred textual or interactive multimedia help-content]

Figure 17: Percentages of users who (A) preferred to search for help textually or vocally, and (B) preferred textual or interactive multimedia help-content.

At the end of satisfaction assessment sessions, the users were asked to judge three overall assessment statements (shown in Figure 18). Most of the users agreed on these statements. The first statement was questioning friendliness of the multimodal environment MMID compared to the visual-only graphical environment TVOID. 79% of the users agreed that MMID was friendlier than TVOID and OFVOID. Friendliness was assessed by the users from different angles. 44% of user views stated that MMID was friendlier because it was easier to use, 31% of them stated it was friendlier because it was faster for accomplishing tasks, 13% of the answers were that MMID was funnier and more exciting to use. Other views (6%) considered MMID friendlier because it was more interactive. In addition, 4% of the answers stated that the vocal commands were easier to learn, and 2% of them considered that MMID was friendlier because it implied less stress on sight. 10% of the users disagreed with this statement because they were worried about sensitivity of MMID toward sounds and accurate pronunciation of commands. One of the opinions stated that "MMID enforces the user to memorize commands and therefore the visual metaphors would be easier to use". Another opinion raised a worry of that "Using an environment like MMID would negatively influence the user technical and administrative skills and cause them to decrease due to limited interaction with textual and graphical metaphors". Figure 18 also shows that 11% of the users (9 users) were uncertain that MMID was friendlier than TVOID as they thought that both environments were friendly and as friendly as each other.

The second statement stated that MMID was easier to learn than the TVOID. 78% of the users agreed on this statement. 28% of the justification answers of why these users considered MMID easier to learn were that because MMID was easier to use. 26% of the user-views considered MMID easier to learn because it offered an easier way to search for help (i.e. searching by voice). 16% of the justification answers stated that MMID was easier to learn because it offered a more interactive environment for interaction. More precisely, 12% of the answers stated that MMID was more interactive because it involved more than one sense for interaction (the visual and auditory senses). Another group of the answers (10%) considered MMID as an easier environment to learn because its vocal commands were gathered in one position within its interface (commands-list), which made it easier to access in case if any of them was forgotten, while on the other hand, TVOID offered many kinds of interaction metaphors that were "scattered" in different locations within the interface. Finally, 8% of the user-views considered MMID easier to learn because it was funnier and more exciting. On the other hand, 9% of the users disagreed with this statement. These users were worried about the difficulty to memorize the vocal commands. One user was worried that the vocal commands were difficult to memorize. These worries rose because these did not speak English as a native language. Two users stated that MMID was new to them; therefore, it would not be easier for them to learn it than learning the familiar graphical metaphors offered in TVOID. Figure 18 also shows that 13% of the users were uncertain that MMID was easier to learn than TVOID because they were worried about the difficulty of memorizing all vocal commands.

The last assessment statement aimed at taking users’ views of whether or not they will use a multimodal interface-design environment if it was produced by Microsoft under the name of Multimodal Studio and stop using the famous Visual Studio which is also produced by the same corporation. Figure 18 shows that 88% of the users agreed on this statement. 29% of these users anticipated that the multimodal studio would be more interactive than the currently existing visual studio because it would provide with more than one way of
interaction (graphics, text, vocal instruction, and speech). 26% of the users thought that the multimodal studio would be easier to use, and 6% of them thought it would be easier to learn because the environment would provide with the features of searching for help by voice and learning interactively. One user stated that he did not like the visual studio because its interface was complex and unfriendly. 18% of the users thought that the multimodal studio would be more efficient than the visual studio in terms of reducing task accomplishment time. 17% of them anticipated that they would move to the multimodal studio because it would be more exciting and more motivating to use than the visual studio. Finally, one user anticipated that it would apply less stress on sight because of the less graphical metaphors it would offer in its interface. On the other hand, Figure 18 also shows that 2% of the users refused the idea of using the multimodal studio instead of the visual studio. One user stated that he preferred visual interaction only because he did not like to hear any sounds from the computer while working. Another user stressed the importance of using the visual (graphical) metaphors while designing interfaces and that they can not be neglected. Figure 18 also shows that 10% of the users were undecided about this statement. One user wrote that "The multimodal studio would not be suitable for the users with hearing and speaking disabilities, and that the visual studio would be more suitable for them".

V. CONCLUSION

This paper introduced an empirical multi-group study for investigating usability of two experimental IDE toolkits (TVOID and MMID) offering visual (graphical) and multimodal (visual and auditory) interaction metaphors. The auditory interaction metaphors investigated by this study was speech as input and output means. Usability of these two environments was measured in terms of efficiency, effectiveness and satisfaction. Efficiency and effectiveness were measured over two experimental phases by independent groups of novel and experienced users. Then, satisfaction was measured dependently by one group of users.

Results from first experimental stage, which was carried out by novel users showed that using speech for input and output have helped MMID to be more efficient and more effective than TVOID in terms of task-accomplishment time and percentage of tasks completed successfully. However, it was less efficient in terms of frequency of errors due to unfamiliarity with the environment which, mainly, caused the users to be less confident when speaking the required vocal commands. This deficiency in confidence caused words and phrases to sound improperly, and consequently, to be either unrecognized or misrecognised by the system.

In the second experimental phase, this problem was solved by increasing the level of confidence-factor in the system such that a spoken command would not be executed unless if it was uttered in an accepted level of confidence. In addition, recruiting experienced (well-trained) users for the experiments insured a high level of familiarity with the system. These two factors helped in reducing the frequency of errors dramatically. On the other hand, it was concluded that using a graphical metaphor very frequently would make the user become careless or hasty during interaction with it, which would cause usability problems to occur.

Outcomes from the third experimental phase revealed how much satisfactory each of the assessed interaction metaphors was for activating menu-command functionality, drawing, setting properties and using help. Most of the users showed enthusiasm toward using vocal instruction in MMID and chose this interaction metaphor as the most preferred for doing all kinds of jobs.

Using speech for completing tasks in MMID has saved the time taken by the users of TVOID in finding hidden menu-items, learning about iconized toolbar-buttons and scrolling for far tools in toolbox and properties in properties-table. It can be concluded that using this interaction metaphor along with limited use of the mouse can effectively replace most of the graphical interaction metaphors that crowd the user-interface generally and IDE interfaces specifically.

REFERENCES


Mohammad Alsuraihi is a software designer and programmer with an interest in human-computer interaction and software engineering; he is a PhD student in the Department of Computing at the University of Bradford.

Dimitris Rigas is a computer scientist with an interest in human-computer interaction, mediated systems for learning, and software engineering; he is a senior lecturer in the Department of Computing at the University of Bradford.