Evaluating Physical/Virtual Occlusion Management Techniques for Horizontal Displays

Waqas Javed, Kyungtae Kim, Sohaib Ghani, Niklas Elmqvist

Evaluating Physical/Virtual Occlusion Management Techniques for Horizontal Displays

Waqas Javed, KyungTae Kim, Sohaib Ghani and Niklas Elmqvist

School of Electrical & Computer Engineering,
Purdue University, West Lafayette, IN 47907-2035, USA
{wjaved | kimk | sghani | elm}@purdue.edu

Abstract. We evaluate unguided and guided visual search performance for a set of techniques that mitigate occlusion between physical and virtual objects on a tabletop display. The techniques are derived from a general model of hybrid physical/virtual occlusion, and take increasingly drastic measures to make the user aware of, identify, and access hidden objects—but with increasingly space-consuming and disruptive impact on the display. Performance is different depending on the visual display, suggesting a tradeoff between management strength and visual space deformation.

1 Introduction

As digital tabletop displays are becoming increasingly inexpensive, common, and available to the mass market, they will also start to be integrated into everyday work environments. In fact, such devices may even totally replace standard desktop computers in the future; as a case in point, Wigdor et al. [1] describe the use of a tabletop display for a single user over the period of a year. However, tabletop displays are horizontal surfaces, and such surfaces invite placing physical objects on them [2], such as paper, books, and even coffee mugs. This will inevitably give rise to problems where these physical objects occlude virtual objects on the tabletop display (Figure 1).

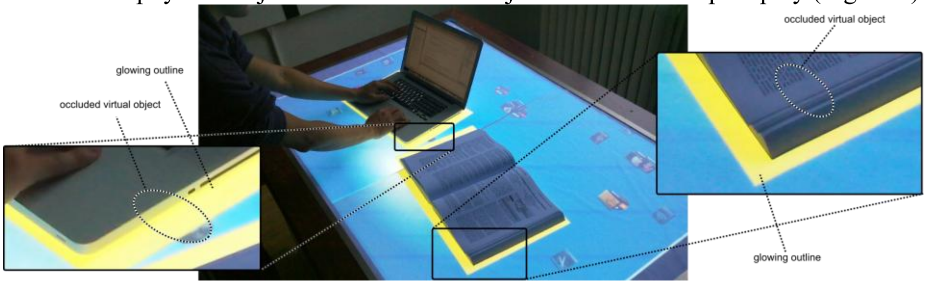


Fig. 1. Tabletop environment with physical objects (a laptop computer and a book) resting on its surface. The physical objects are occluding virtual objects on the display, so their outlines have been made to glow using one of our occlusion management techniques to indicate this.

Mitigating occlusion is not a new problem and has been addressed previously: for 2D desktop environments, hidden overlapped windows can be retrieved using the taskbar in Microsoft Windows or the dock in Mac OS X. More advanced techniques make windows semi-transparent [3], cut holes in them to expose hidden items [4], or spread them out to make hidden content accessible [5]. Other work considers situations when the user's hand and arm occlude the display on touch and pen-based systems [6–8]. However, with the exception of contextual display bubbles [9] and relations between virtual and digital media [2], little work has been performed on managing occlusion between virtual and physical objects beyond the user's own body.

We propose a framework for occlusion management in such hybrid physical/virtual environments based on three levels of occlusion management: awareness (knowledge that a particular target exists), identification (target is recognizable but cannot be interacted with), and access (the ability to fully interact with the target). These levels, together with additional characteristic parameters, define the design space of tabletop occlusion management. Using this design space, we derive six techniques that represent the spectrum of occlusion management: adding glow to the outlines of physical objects (GLOW, see Figure 1), displaying miniature icons of occluded objects (ICON), creating hybrid piles of virtual and physical objects (PILE), introducing a small overview map (MMAP), displacing virtual objects to empty space (MOVE), and, finally, replicating hidden area in unoccluded space (REPL).

Our evaluation of these techniques begins with a formative study intended to elicit requirements and constraints for a more in-depth evaluation. We then present results from two quantitative user studies designed to determine which technique was most efficient in managing occlusion. Both studies involved 12 (different) participants who performed unguided (Experiment 1) and guided (Experiment 2) visual search in the presence of both physical and virtual distracting objects. For Experiment 1, MOVE was the fastest technique, but the situation is reversed in Experiment 2, where displacing objects has a negative impact. We speculate that a combination is optimal.

2 Related Work

Existing research has studied occlusion management for virtual objects, for touch and pen-based interaction, and for tabletop displays. Below we review these in detail.

2.1 Virtual Occlusion

Windows often overlap in 2D desktop systems. To interact with an overlapped window, users must move, resize, or change the stacking order of windows. To avoid this problem, window managers typically provide ways to cascade or tile windows [10]. A window index, like the Windows taskbar or the MacOS Dock, allows for directly accessing windows even if they are hidden behind other windows. Another approach is to mitigate occlusion by placing windows in regions with the least overlap [11].

Alpha blending [12] can be used to show the content of occluded windows using transparency. The problem with this technique is that it is difficult to visually determine which content belongs to which window. Multi-blending [3] addresses this issue through a framework of transparency filters to allow simultaneous display of transparent windows. Free-space transparency [4] selectively turns unimportant window regions transparent and important regions opaque to optimize visibility.

2.2 Physical Body Occlusion

Occlusion in both direct touch and pen-based interaction arises when the user's own hand or arm covers some portion of the display. Ample research has shown that this occlusion is problematic and can severely impede performance [13, 14]. Forlines et al. [15] found that occlusion causes decay in performance in one-dimensional tapping and crossing tasks. Supporting this finding, Leithinger and Haller [16], while investigating menu placement for tabletops, discovered that menus suffering from occlusion had significant disadvantages compared to other menus. In fact, Vogel et al. [8] found that the user's hand and forearm can occlude up to half of a 12-inch display.

Various interaction techniques have been designed with occlusion in mind [17–19]. Hancock et al. [13] improve awareness in menu placement for tabletop display by designing menus that detects user handedness. Brandl et al. [20] also designed occlusion-aware menus that are positioned around the table so that they are not occluded.

Some techniques actively mitigate occlusion for touch displays. Ramos and Balakrishnan [18] use a sinusoidal-shaped slider to reduce occlusion from the user's hand in pen-based displays. CrossY [17] uses right-to-left movement to reduce occlusion with right-handed users. Shen et al. [21] utilize a popup widget to overcome occlusion caused by the user's finger for touch displays. This is similar to Shift [7], where a copy of the occluded region is moved to a non-occluded region. Building on this, more recent work on occlusion-aware interfaces [6] models which area is occluded and uses this knowledge to display hidden objects in a non-occluded screen area.

2.3 Physical/Virtual Occlusion

The environment-aware display bubbles presented by Cotting and Gross [9] are perhaps most relevant to our work. Their technique projects structured light onto the table to acquire inadmissible areas, which are then used to deform the display bubbles.

Recent work by Steimle et al. [2] studied usage patterns of physical and virtual media on tabletop surfaces. They found the expected drawbacks of physical occlusion, such as information hiding and interaction costs, but also—interestingly—some desirable features for organizing and partitioning information (particularly for hybrid piles [22]). Their results indicate the need for tabletop interfaces to explicitly support physical interaction on, above, and around the surface, further motivating our work.

3 Motivation: Everyday Use of Tabletops

Already in 1991, Shneiderman [23] listed the advantages of touch displays and predicted that, as more research will overcome their disadvantages, they will become more and more popular. In 1993, Wellner [24] first described the idea of an office desk as a digital display. True to form, tabletops are nowadays becoming increasingly available to the mass market, and are starting to appear in showrooms, lobbies, and museums. With the decreasing cost of large display technology and increasing demand from a growing potential consumer base, the step to adoption in the living room—and certainly the office—cannot be far away. In a pioneering effort, Wigdor et al. [1] describe the use of a tabletop for long-term office use and its pros and cons.

Following this reasoning, it is plausible that in the future, digital tabletops will be commonly used as office desks—or rather, that many of our horizontal surfaces in offices (such as desks) will be appropriated as displays [25]. However, the physical affordance of horizontal surfaces also invite placing various physical items on them [2], such as coffee mugs, papers, books, etc. When the desk is also a digital display, these items will start to hide virtual objects on the display.

4 Occlusion Management on Tabletop Displays

We define *tabletop occlusion management* as methods that give awareness, identification, or access to virtual objects that are occluded by physical objects on tabletops.

4.1 Detecting Physical Occlusion

The first step towards *managing* physical occlusion is to *detect* it. For tabletop displays built using diffuse illumination (DI) [26], this is relatively easy. DI tabletops have active infrared emitters that cause reflections on a physical object resting on the tabletop surface, allowing the camera tracking system to detect its actual outline. Taken together, these outlines represent occluded space on the display, and become the input for all types of occlusion management techniques.

The single remaining difficulty is for tabletop systems that already rely on DI technology for the touch interaction (not all do—for example, SLAP widgets [27] use FTIR [28] for touch and DI for detecting objects), forcing us to distinguish between physical occluders and the hands of the users. We suggest using a heuristic based on the motion of an object—if an object has not moved for T (e.g., 10) seconds, we regard it as an occluder. Other approaches would be to require a minimum size for an object to be classified as an occluder, or to put fiducial markers on the undersides of physical objects on the display (the latter is not a general solution, however).

Finally, we should note that physical objects are three-dimensional, and thus also have a geometric extension above the table surface. This means that accurate occlusion detection should consider the line of sight of the user, whereas our implementation only utilizes the 2D footprint of an object on the tabletop surface. However, our techniques would also work with a line-of-sight based approach, and thus the technical limitations with our implementation do not affect the validity of our results. We discuss this aspect in more detail in the design implications section.

4.2 Levels of Occlusion Management

We define four levels of increasing occlusion management:

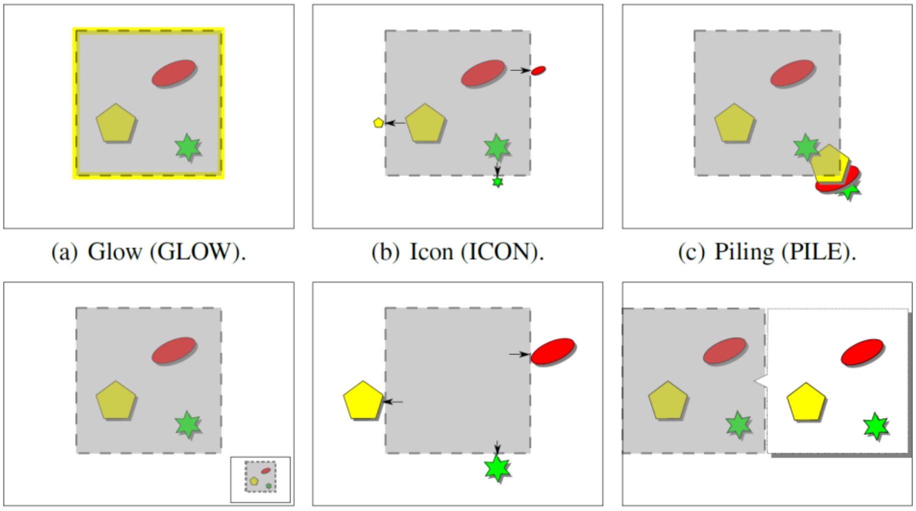
- **No knowledge**: A virtual object is fully occluded by a physical object, and the user is unaware of its existence. The virtual object will remain hidden until the user happens to move or lift the occluding object.

- **Awareness**: A user that is aware of an occluded virtual object knows that one or more virtual objects are occluded by a physical object, but does not necessarily have knowledge about the number or identity of the hidden objects.
- **Identification**: A user that can identify an occluded virtual object can see (part or a copy of) the virtual object, but cannot interact directly with the object. To interact with the object, the user must lift or move the occluding physical object.
- **Access**: Having access to an occluded virtual object means that the user is able to fully view and interact with the object as if it was not hidden at all.

4.3 Design Parameters

We suggest four additional design parameters for classifying occlusion management:

- **Space Consumption**: Amount of space utilized by the technique in addition to the space utilized by the physical objects themselves.
- **Visual Clutter**: Degree of visual clutter produced. With occluded objects no longer visible, how much new clutter is introduced by the technique?
- **Spatial Transformation**: Impact on the geometric properties of the visual space, including distortion, displacement, and rotation.
- **Intention Support**: Intentional occlusion possible. The user may want to intentionally hide virtual objects using physical objects. Is this scenario possible?



(a) Glow (GLOW). (b) Icon (ICON). (c) Piling (PILE).

(d) Minimap (MMAP). (e) Move (MOVE). (f) Replicate (REPL).

Fig. 2. Overview of occlusion management techniques evaluated in this paper. The large gray rectangle represents the footprint of a physical object placed on a tabletop display, thus occluding three virtual objects drawn on the actual display (ellipse, star, and pentagon).

4.4 Techniques

We derive six techniques for different levels of occlusion management. In Figure 2, we use a standard situation with three virtual objects (an ellipse, star, and pentagon) being occluded by a single physical object placed on the tabletop surface.

- **Awareness-supporting**: This type of technique only makes the user aware of occluded virtual objects without actually showing *which* objects are occluded. For this reason, such techniques take the least amount of visual space of all proposed techniques. They also respect the user's intention when hiding an object.
  - *Glow (GLOW)*: Displays a colored glow around the outline of physical objects that are occluding one or more virtual objects (Figure 2(a)).
- **Identification-supporting**: These techniques provide not only awareness, but also allow users to identify the occluded virtual objects. For this reason, the space consumption and visual clutter is often higher than for awareness-supporting techniques. However, while these techniques typically give hints about the occluded content (through miniatures or portions of the objects), the user is still required to manipulate the physical object to interact with the virtual objects.
  - *Icon (ICON)*: This technique displays a fixed-size small visual representation—i.e., an icon—of an original occluded object (Figure 2(b)). The icon will be displayed on the nearest edge of the occluding physical object in order to convey some information on where the hidden virtual object is placed.
  - *Piling (PILE)*: This technique mimics a traditional office desk where people pile papers and books into neat stacks (Figure 2(c)). The technique organizes hidden virtual objects into piles under the occluding physical object, thus showing a small portion of the original objects as a hint to the user (similar to [22]).
  - *Minimap (MMAP)*: The *minimap* technique shows a miniature overview of the virtual space with physical objects outlined (Figure 2(d)). The user can refer to the overview to find the location of hidden objects.
- **Access-supporting techniques**: This family of techniques gives the user full input and output access to hidden virtual objects. To make this possible, these techniques have high space consumption and visual clutter. However, the result is an entirely occlusion-free visual space with no physical object interaction necessary; on the other hand, users cannot intentionally cover virtual objects with physical items.
  - *Move (MOVE)*: Moves hidden objects to the nearest edge of its occluding object (Figure 2(e)). Items can be connected to their original positions using a virtual spring, causing them to spring back when the occluder is removed.
  - *Replicate (REPL)*: Inspired by Vogel et al. [6], this technique replicates the entire screen content of a particular physical occluder in a callout located in unoccluded space as close to the occluder as possible (Figure 2(f)). Objects are revealed in their natural size and their relative position is maintained.

We do not claim that the above six techniques are exhaustive examples of the design space of physical occlusion management; additional possibilities include object indices, distortion, and labels, etc. We also do not claim that these techniques are novel; in fact, some explicitly derive from existing techniques such as Shift [7], hybrid piles [22], and occlusion-aware interfaces [6]. However, we do think these techniques are a representative sampling of the occlusion management design space.

5 Designing the Evaluation

Because our work is the first (to the best of our knowledge) to study techniques for managing occlusion between physical and virtual objects on a tabletop display, we wanted to perform a canonical evaluation of the problem. To inform the design of this evaluation, we first conducted a formative evaluation involving graduate students from our university. Below we discuss the design decisions made based on this study.

5.1 Evaluation Platform

For the purposes of evaluating the above techniques, we implemented a graphical framework in Java for multitouch tabletop displays based on the Piccolo 2D graphics API and the TUIO [29] touch event protocol. The framework detects both finger touches as well as physical objects on the surface—an object that has not moved for a given time (10 seconds, in our case) is regarded as an occluder. We encourage the reader to view the paper's video for footage of our occlusion management software.

5.2 Formative Evaluation

We conducted a formative evaluation of the testing platform with three participants. All participants performed a jigsaw puzzle task where they reassembled a picture split into 3×3 tiles back into the original image. The tiles were randomly placed on the tabletop surface but were not rotated (in fact, tiles could only be moved, not rotated). We also randomly placed four sheets of paper (physical occluders) on the tabletop surface. Participants performed one reassembly task for each of our techniques.

Because of the formative nature of this evaluation, we did not collect any correctness or time measurements. However, we took careful notes of participant behavior. The qualitative feedback was very positive despite tracking inaccuracies inherent to our tabletop. While the jigsaw puzzle task with sheets of paper as occluders worked well, we found it difficult to control the placement of the occluders on the tabletop surface. Furthermore, even though we asked participants not to pile or remove occluders from the display, a common strategy was to initially move them out of the way before solving the task, thus neutralizing the impact of any specific occlusion management technique. This was an issue we needed to address in the full experiment.

5.3 Design Decisions

Based on the formative evaluation, we made the following decisions:

- **Low-level visual search tasks**: Given our intent to build a solid foundation for physical/virtual occlusion management, we decided to select fairly low-level tasks involving *visual search* [30] for our evaluation. Because visual search generally is split into guided and unguided search (depending on whether the user has a priori knowledge of the spatial position of the target or not), we decided to perform two separate experiments, one for each of these. We argue that these tasks are building blocks for higher-level tasks involving more complex objects like text documents and applications, so our results should therefore generalize even to such objects.

- **Static occlusion**: Existing work [6, 8] on physical occlusion between a user's arm and the display deals with intrinsically dynamic situations with moving targets and occluders. However, the primary problem on tabletops is *static* occlusion caused by physical objects placed on the table surface. For this reason, and for the sake of simplicity, we do not consider dynamic situations in our evaluation. Nevertheless, our occlusion management techniques handle dynamic occlusion equally well as static occlusion, so we think that the results should again generalize to other settings. In fact, our framework implementation requires physical objects to be static on the tabletop surface for at least 10 seconds to be regarded as occluders. Without being able to otherwise differentiate interaction touches and occluders, we think is a reasonable strategy for realistic occlusion management implementations.
- **Small targets**: Occlusion can occur at many scales (such as for small icons vs. large documents) as well as on many levels (i.e., partial vs. total occlusion). On the basis that large and complex visual objects can generally be decomposed into many smaller objects, we opted to design our tasks for small and generic targets.
- **No physical objects**: Based on our experience from the formative evaluation, we chose not to include any actual physical objects in this study, but instead simulated them using virtual objects placed in the top layer of the visual space. This enabled us to fully control the placement of physical occluders for the purposes of the experiment. While this is naturally counterintuitive in a study on physical/virtual occlusion, we do not believe this affects the validity of our results in any way—the efficacy of the techniques themselves are independent of the source of occlusion.
- **Immovable occluders**: Again drawing on our observations from the formative evaluation, we decided not to allow occluders to be moved to avoid having users spend the first portion of each trial moving items to one side of the visual space. While this is clearly an artificial constraint imposed for mostly technical reasons, we can defend this by the fact that in many cases, it is either not practical to move objects away to clear space for interaction (such as for heavy books, large stacks of paper, keyboard or monitors), or the physical items are intimately tied to the task the user is performing (citing research papers, reading manuals, writing notes, etc).

6 Experiment 1: Unguided Visual Search

Our objective with this study was to derive the properties of our occlusion management techniques for unguided visual search. In perceptual studies, *unguided visual search* is defined as an active scan of a visual environment for a particular target whose location is not previously known [30]. The goal was to study technique performance under different space and object conditions. In particular, we opted **not** to compare techniques to a baseline with no occlusion management because our pilot testing made it obvious that the comparison was in favor of all of the new techniques.

6.1 Apparatus

We conducted the experiment on an 1.2m×0.9m (approximately 81 inch) DI [26] multitouch tabletop display equipped with with two DLP projectors, each with 1280×800 resolution (for a total of 1600×1280). The projectors were powered by a computer running Microsoft Windows 7.

6.2 Participants

We recruited 12 paid volunteers (8 males, 4 females) for the experiment. Ages ranged from 21 to 29 (average 25.3, median 23) years, and all participants had normal or corrected-to-normal vision with no color deficiency (self-reported).

6.3 Task

We designed our experiment as a basic instance of unguided visual search [30], where the participants were asked to find a known *target object* in an unstructured visual space. The visual space consisted of both virtual and physical objects:

- **Virtual objects**: These objects are graphical elements drawn on the display, and include the target as well as a set of *distracting objects* that have a similar visual appearance as the target. More specifically, **all** virtual objects were 3×3 rectangular grids of white cells, with one cell colored red. For the target, the colored cell was always the center one, whereas objects with other positions of the colored cell were distractors (Figure 3 gives an overview). The rationale for this design was that targets should not be preattentively distinguishable from distractors.
- **Physical objects**: These represent *physical occluders* (such as books, stacks of paper, or coffee mugs) placed on the tabletop surface, thereby potentially hiding virtual objects on the display. To allow us to fully control the number and placement of occluders, these were actually **not** physical in our experiment; instead, they were represented by gray rectangles (300×300 pixels in size) placed in a layer above the virtual objects. Tapping on an occluder's rectangle temporarily *lifted* the object to make occluded content visible; tapping again on the screen would return that occluder to its original position. Only one occluder could be lifted at a time. We enforced a 1-second delay between lifts to model real occluders.

The space was configured so that all virtual objects were randomly placed behind a physical occluder, and thus initially occluded. Furthermore, all objects were placed so that they did not overlap (except between virtual objects and occluders) and were within reach of a participant without having to move; however, some techniques like MOVE could potentially cause an object to be moved out of reach or to partially overlap another object. Because of the high number of virtual objects, we could not rely on chance to make some physical occluders not occlude any object, so we controlled this by enforcing one occluder to be a dummy occluder, i.e., with no virtual object underneath. This gave some benefit to GLOW by cutting down search space.

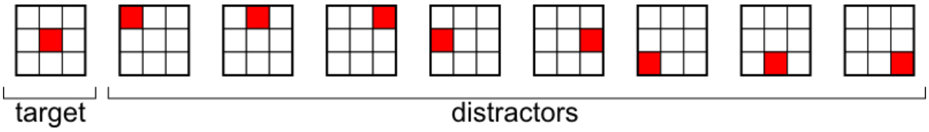


Fig. 3. Virtual objects in Experiment 1. The object on the left is a target, others are distractors.

6.4 Experimental Conditions

We included three factors in our study: Technique, Object Size, and Object Number.

- **Technique**: We included all of the occlusion management techniques above as a factor *T*: GLOW, ICON, PILE, MMAP, MOVE, and REPL.
- **Object Size**: Because identification-supporting techniques often rely on miniature versions of virtual objects, we included Object Size *S* as a factor in the experiment with two levels: *large* and *small*. Our pilot testing helped us set values to these levels: small objects were 42×42 mm (60×60 pixels) and fully recognizable at normal magnification, and large objects were 56×56 mm (80×80 pixels) or about a quarter of physical occluders (210×210 mm or 300×300 pixels). In particular, the small size was selected so that the miniature versions of targets would not be trivially identifiable using the ICON, MMAP, and PILE identification techniques.
- **Object Number**: Techniques with high visual clutter and space consumption are presumably sensitive to the total number of virtual objects in a visual space. To test this hypothesis, we included an Object Number factor *N* with two levels: *few* and *many*. Again, pilot testing helped us find values for these: 10 objects vs. 20.

6.5 Experimental Design

We used a full factorial within-participants design with the following factors:

	12	participants
×	6	**Techniques** *T* (see above)
×	2	**Object Sizes** *S* (Small, Large)
×	2	**Object Number** *N* (Few, Many)
×	10	repetitions (training excluded)
	2880	total trials (240 per participant)

Trials were organized in blocks for each technique. Block order was balanced using a Latin square across participants to counteract learning effects; other factors were randomized within blocks. The experiment platform collected completion time.

6.6 Procedure

Participants received training with each technique before each block until they indicated that they were ready to proceed. Trials were interleaved with an intermission screen during which participants could rest, also ensuring that their hands were in a neutral position prior to each trial. When starting a trial (and the timer), the screen changed to show the visual space with physical and virtual objects. Participants were instructed to complete the task as fast as possible. The trial ended (stopping the timer) when the participant found and tapped the correct target (or its copy in the REPL technique). Tapping on the wrong virtual target made it flash red to indicate the error.

After finishing each technique block, participants were given a structured interview where they were asked about that technique. A full experimental session, including training and post-block structured interview, lasted approximately 60 minutes.

6.7 Hypotheses

H1 Access-supporting techniques (MOVE, REPL) will yield faster performance than identification-supporting techniques (ICON, MMAP, PILE).

H2 Identification-supporting techniques (ICON, MMAP, PILE) will yield faster visual search performance than awareness-supporting ones (GLOW).

6.8 Results

For each trial, we measured the completion time, the number of occluder lifts, and the number of taps on objects. For lifts and taps, we found no significant differences between techniques, and so we disregard these measures for the rest of this analysis.

Figure 4(a) shows the completion times for each technique (three standard deviation outliers removed per participant and condition). The distribution of completion times was not normal, so we analyzed its logarithm using a repeated-measures analysis of variance (RM-ANOVA, normality and homogeneity of variances assumptions checked with Shapiro-Wilks and Bartlett's tests, respectively, and both valid after log-transformation)—see a summary of completion time effects in Table 1.

We used Tukey's HSD test to analyze pairwise differences for techniques on completion time—significant results (*p* < *.05*) are summarized in Figure 5 (left). In particular, MOVE performed significantly faster than any other technique (*p* < *.05*), followed by REPL and GLOW, and finally MMAP, ICON, and PILE as the slowest.

Factors	df, den	F
Technique (*T*)	5,55	**16.53
Object Size (*S*)	1,11	**23.61
Object Num (*N*)	1,11	**60.14
T * *S*	5,55	*3.05
T * *N*	5,55	**5.05
S * *N*	1,11	**17.87
T * *S* * *N*	5,55	0.05

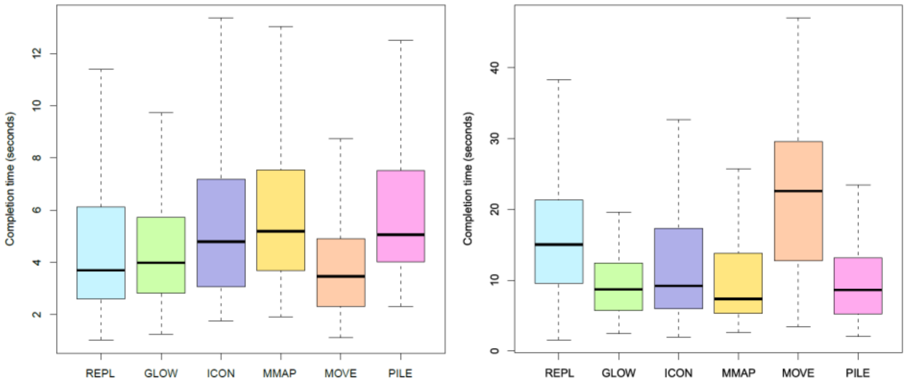
Table 1. Significant effects of factors on time (Experiment 1, RM-ANOVA).

Factors	df, den	F
Technique (*T*)	5,55	**23.29
Occl. degree (*O*)	1,11	0.025
T * *O*	5,55	2.12

Table 2. Significant effects of factors on time (Experiment 2, RM-ANOVA).

Significance results:

*	$= p \leq .05$
**	$= p \leq .0001$



(a) Experiment 1. (b) Experiment 2.

Fig. 4. Completion times for occlusion management techniques for both experiments.

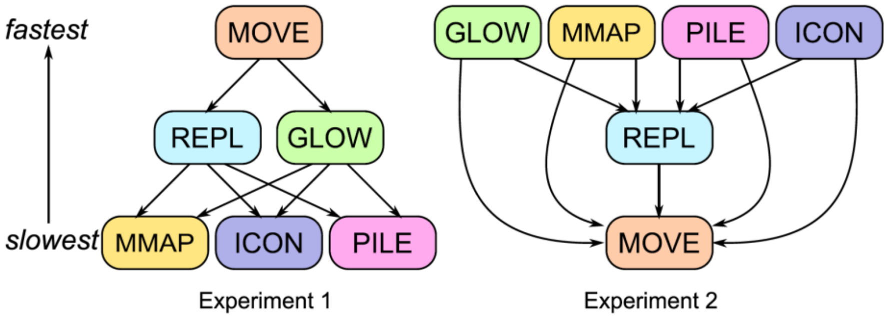


Fig. 5. Significant time differences (*p* < *.05*) for techniques in Experiment 1 (left) and 2 (right).

6.9 Discussion

Our results show that for unguided visual search, MOVE and REPL perform significantly better than ICON, PILE, and MMAP. While we have only tested a sample of the occlusion management design space, we think that this means that access-supporting techniques (MOVE, REPL) perform better than identification-supporting techniques (ICON, PILE, MMAP) for unguided visual search. This confirms **H1**, and it is clearly due to the fact that MOVE and REPL make all targets visible and selectable without requiring any occluders to be lifted. Also, the drawback of these techniques—displacing objects from their original positions—had no impact on the nature of the task. Identification-supporting techniques, on the other hand, generally show a smaller version of the targets, making recognition more difficult.

For **H2**, we were surprised to find that GLOW performs significantly better than ICON, PILE, and MMAP, thus rejecting the hypothesis. This may partially be due to the reason given above: recognizing the target is harder for these techniques. With

GLOW, participants immediately started lifting the occluders to find the target object, whereas for ICON, PILE, and MMAP, they would essentially have to perform two visual search tasks: first for miniature versions on the whole space, and then again under the occluder they lifted. This was also supported by feedback from interviews.

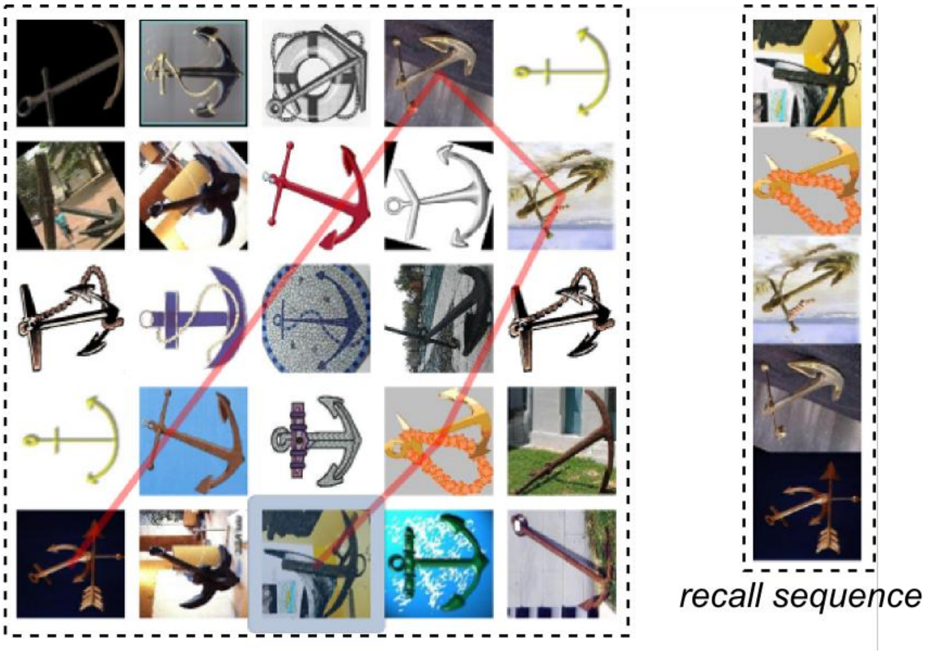


Fig. 6. Typical scenario for Phase I in Experiment 2.

7 Experiment 2: Guided Visual Search

Beyond the pure unguided visual search in the first experiment, we wanted also to study the impact of occlusion management for *guided visual search* [30] where the spatial location of objects is important. We thus designed Experiment 2 as a game of Concentration with face-up photographs. Beyond task, we used the same apparatus and procedure as in Experiment 1. Below we discuss differences between the studies.

7.1 Participants

We recruited 12 paid volunteers (10 males, 2 females) for the experiment. Ages ranged from 22 to 28 (average 24.3, median 23) years, and all participants had normal or corrected-to-normal vision with no color deficiency (self-reported).

7.2 Task

Guided visual search tasks require that the participants have some a priori knowledge of target location. We achieved this using a revisitation task consisting of two phases:

I. **Learning**: Users were shown a 5×5 grid of photographs (each 100×100 mm, or 150×150 pixels, in size and visible at all times). A randomly selected photograph would be highlighted. By tapping and holding the highlighted photograph for a minimum *dwell time* (1.5 seconds), the highlight would move on to a new photograph. This was repeated 5 times. We also showed a red path connecting all of the photographs to visit to further aid participants in learning the locations.

II. **Recall**: After having visited (and presumably memorized) the location of 5 photographs, the user was asked to revisit the photographs (still face-up and visible) in the recall phase. During this time, a set of physical occluders (again simulated using gray rectangles) was added to the display in random locations, thus obscuring part of the grid. As before, occluders could be lifted, one at a time, by tapping. We enforced the photograph sequence from the learning phase, so participants could only proceed to the next photograph in the sequence by selecting the correct photograph (i.e. no out-of-order selections were possible).

Figure 6 shows a typical scenario during Phase I (learning).

We timed participants in Phase II, but there was no time limit on Phase I. The sequence of photographs was shown on the right side of the screen with the next photograph to visit highlighted, and out-of-order selections were not possible. We did this to eliminate errors in the experiment, allowing us only to analyze completion time.

Finally, photographs for each trial were selected from the same random category of the CalTech-101 image dataset [31]; this ensured that all images had the similar motif. We manually eliminated images that were almost identical, however.

7.3 Experimental Conditions

We included two factors: Technique *T* (same as Experiment 1) and Degree of Occlusion *O*. The degree of occlusion was simply the number of physical occluders in Phase II (recall).We used the same size occluders as before (210×210 mm or 300×300 pixels) and included two levels: *low* (1 occluder) and *high* (2).

7.4 Experimental Design

We used a full factorial within-participants design with the following factors:

	12	participants
×	6	**Technique** *T* (see above)
×	2	**Degrees of Occlusion** *O* (Low, High)
×	4	repetitions (training excluded)
	576	total trials (48 per participant)

As before, trials were organized in blocks for each technique, and block order was counterbalanced using a Latin square (*O* randomized). The experimental platform collected completion time as well as the number of taps and lifts per trial.

7.5 Hypothesis

H3 Identification-supporting techniques (ICON, MMAP, PILE) and awareness-supporting techniques (GLOW) will yield faster guided visual search performance than access-supporting techniques (MOVE, REPL).

7.6 Results

Figure 4(b) shows the completion times for each technique. We again observed that completion time measurements did not obey a normal distribution, so we analyzed its logarithm using an analysis of variance (RM-ANOVA, normality and homogeneity of variances assumptions checked as above and valid after log-transformation) to find significant effects of the factors on completion time—see Table 2 for results.

To compare techniques directly, we performed a Tukey's HSD posthoc test. Results of this analysis can be found in Figure 5 (right). In particular, all other techniques were significantly ($p < .05$) faster than REPL and MOVE, and REPL was in turn significantly ($p < .05$) faster than MOVE (almost a reversal from Experiment 1).

7.7 Discussion

The results from Experiment 2 confirm hypothesis **H3**—techniques with less impact on the spatial position of targets performed better than access-supporting techniques that drastically affect the spatial arrangement on the screen. This is not a surprising result given that guided visual search relies on a priori knowledge of target locations.

Our results also show that REPL performed significantly faster than MOVE, indicating that the design of access-supporting techniques is an important factor to consider. In this case, the fact that REPL maintains the relative positions of objects in the locality of occluded space is probably the reason for this difference in performance. In other words, both MOVE and REPL transform the absolute positions of objects, but REPL at least maintains relative positions.

8 Implications for Design

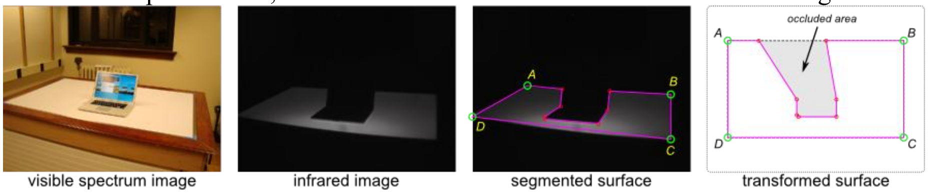
Based on our findings from both of the experiments, there is no one technique that performs optimally for both guided and unguided visual search. There is clearly a tradeoff between a technique's impact on virtual objects on the display, and its capability to mitigate occlusion due to physical objects. In such circumstances, a balance must be struck between these two factors. The GLOW technique performs only second to MOVE in Experiment 1, and is among the fastest techniques in Experiment 2. It is also relatively lightweight to implement and does not significantly alter the interaction in existing applications. For these reasons, we recommend the use of GLOW as a starting point for managing physical/virtual occlusion on tabletops.

Having said that, there is nothing that prevents us from combining techniques to achieve better performance. For example, the minimap (MMAP) technique is a fairly nonintrusive addition to any application, and may actually serve additional purposes for overview+detail in maps, visualizations, and graphical editors. It is also true that these techniques represent only a sample of the occlusion management design space. Although we based our designs on a systematic exploration of this space, it is possible that other designs exist that would perform better. This is a topic for future research.

Furthermore, our experiments tested only low-level visual search, but general tabletop applications consist of many other types of tasks at different abstraction levels, as well as potential targets of many different shapes and sizes. However, we argue that visual search is a central activity in interactive computing, so our results should generalize to many real-world tabletop applications. Our results for relatively small objects should also generalize to components of larger objects. Still, it would be interesting to study occlusion management in longitudinal and more realistic settings.

8.1 Detecting 3D Occlusion

This work does not consider the fact that there is more to physical objects than their 2D footprint on the surface of the tabletop display, and that physical objects also extend along the vertical axis. This means that objects may be hidden from the user's viewpoint by physical objects even if the display area they inhabit is not actually covered—in other words, the occlusion problem between physical and virtual objects on tabletop displays is 3D and not 2D. Provided that we had full knowledge of the visible and hidden areas of the display from the user's viewpoint, we could adapt our techniques to also work in this setting. Our work is a first step towards fully occlusion-aware tabletop interfaces, but more research is needed to reach this ultimate goal.



visible spectrum image
infrared image
segmented surface
transformed surface

Fig. 7. 3D occlusion detection process using an infrared camera mounted on the user's head.

We have experimented further towards reaching this goal by mounting an infrared webcam on a headset so that the camera sees what the user sees (similar to Vogel [8]). With a DI tabletop, the surface of the table will glow with infrared light—except in places where the surface is occluded from the user's point-of-view (Figure 7). This allows us to separate the visible area of the tabletop as a 2D shape. By transforming the 2D shape of visible surface to the coordinate space of the table, we can derive occlusion information that can be used as input for any of the techniques proposed here.

9 Conclusion and Future Work

We have presented a novel approach to manage occlusion between physical items resting on tabletop displays and virtual objects projected on the display. Our primary objective is not to find an optimal occlusion management technique, but rather to characterize the performance of different techniques under different spatial characteristics and tasks. Results from two user studies on guided and unguided visual search, respectively, clearly show that the different design alternatives have different strengths and weaknesses, and one outcome from our work is perhaps that the more lightweight approaches are more promising for the general case.

In the future we plan on continuing to study occlusion management for tabletop displays. Much work remains to be done for detecting physical/virtual occlusion in 3D. We also envision techniques that distort the visual space to aid visual search.

Acknowledgements

This material was partially supported by the U.S. Department of Homeland Security's VACCINE Center under award no. 2009-ST-061-CI0001.

References

1. Wigdor, D., Penn, G., Ryall, K., Esenther, A., Shen, C.: Living with a tabletop: Analysis and observations of long term office use of a multi-touch table. In: Proceedings of the IEEE Workshop on Tabletop Displays. (2007) 60–67
2. Steimle, J., Khalilbeigi, M., Mülhäuser, M., Hollan, J.D.: Physical and digital media usage patterns on interactive tabletop surfaces. In: Proc. ACM ITS. (2010) 167–176
3. Baudisch, P., Gutwin, C.: Multiblending: displaying overlapping windows simultaneously without the drawbacks of alpha blending. In: Proc. ACM CHI. (2004) 367–374
4. Ishak, E.W., Feiner, S.K.: Interacting with hidden content using content-aware free-space transparency. In: Proc. ACM UIST. (2004) 189–192
5. Ramos, G., Robertson, G., Czerwinski, M., Tan, D., Baudisch, P., Hinckley, K., Agrawala, M.: Tumble! Splat! Helping users access and manipulate occluded content in 2D drawings. In: Proc. ACM AVI. (2006) 428–435
6. Vogel, D., Balakrishnan, R.: Occlusion-aware interfaces. In: Proc. ACM CHI. (2010) 263–272
7. Vogel, D., Baudisch, P.: Shift: a technique for operating pen-based interfaces using touch. In: Proc. ACM CHI. (2007) 657–666
8. Vogel, D., Cudmore, M., Casiez, G., Balakrishnan, R., Keliher, L.: Hand occlusion with tablet-sized direct pen input. In: Proc. ACM CHI. (2009) 557–566
9. Cotting, D., Gross, M.H.: Interactive environment-aware display bubbles. In: Proc. ACM Symposium on User Interface Software & Technology. (2006) 245–254

10. Bell, B., Feiner, S.: Dynamic space management for user interfaces. In: Proc. ACM UIST. (2000) 239–248
11. Bell, B., Feiner, S., Höllerer, T.: View management for virtual and augmented reality. In: Proc. ACM UIST. (2001) 101–110
12. Porter, T., Duff, T.: Compositing digital images. Computer Graphics, 18(3) (1984) 253–259
13. Hancock, M.S., Booth, K.S.: Improving menu placement strategies for pen input. In: Proc. GI. (2004) 221–230
14. Inkpen, K., Dearman, D., Argue, R., Comeau, M., Fu, C.L., Kolli, S., Moses, J., Pilon, N., Wallace, J.R.: Left-handed scrolling for pen-based devices. International Journal of Human Computer Interaction 21(1) (2006) 91–108
15. Forlines, C., Balakrishnan, R.: Evaluating tactile feedback and direct vs. indirect in pointing and crossing selection tasks. In: Proc. ACM CHI. (2008) 1563–1572
16. Leithinger, D., Haller, M.: Improving menu interaction for cluttered tabletop setups with user-drawn path menus. In: Proc. IEEE Tabletop. (2007) 121–128
17. Apitz, G., Guimbretière, F.: CrossY: a crossing-based drawing application. ACM Transactions on Graphics 24(3) (July 2005) 930–930
18. Ramos, G., Balakrishnan, R.: Fluid interaction techniques for the control and annotation of digital video. In: Proc. ACM UIST. (2003) 105–114
19. Zeleznik, R.C., Miller, T.: Fluid inking: augmenting the medium of free-form inking with gestures. In: Proc. GI. (2006) 155–162
20. Brandl, P., Leitner, J., Seifried, T., Haller, M., Doray, B., To, P.: Occlusion-aware menu design for digital tabletops. In: Ext. Abstr. of ACM CHI. (2009) 3223–3228
21. Shen, C., Hancock, M.S., Forlines, C., Vernier, F.D.: CoR2Ds: Context-rooted rotatable draggables for tabletop interaction. In: Proc. ACM CHI. (2005) 1781–1784
22. Khalilbeigi, M., Steimle, J., Mühlhäuser, M.: Interaction techniques for hybrid piles of documents on interactive tabletops. In: Ext. Abstr. of ACM CHI. (2010) 3943–3948
23. Shneiderman, B.: Interface—touch screens now offer compelling uses. IEEE Software 8(2) (March 1991) 93–94, 107
24. Wellner, P.: Interacting with paper on the DigitalDesk. CACM 36(7) (1993) 86–96
25. Harrison, C.: Appropriated interaction surfaces. Computer 43(6) (2010) 86–89
26. Çetin, G., Bedi, R., Sandler, S., eds.: Multitouch Technologies. NUI (2009).
27. Weiss, M., Wagner, J., Jansen, Y., Jennings, R., Khoshabeh, R., Hollan, J.D., Borchers, J.O.: SLAP widgets: bridging the gap between virtual and physical controls on tabletops. In: Proc. ACM CHI. (2009) 481–490
28. Han, J.Y.: Low-cost multi-touch sensing through frustrated total internal reflection. In: Proc. ACM UIST. (2005) 115–118
29. Kaltenbrunner, M.: reacTIVision and TUIO: A tangible tabletop toolkit. In: Proc. ACM ITS. (2009) 9–16
30. Treisman, A.: Search, similarity, and integration of features between and within dimensions. Journal of Experimental Psychology: Human Perception & Performance 17(3) (1991) 652–676
31. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In: Proc. Workshop at IEEE CVPR. (2004)