Finding Layers Using Hover Visualizations

Evan Shimizu[*] Carnegie Mellon University

Matt Fisher[†] Adobe Research

Sylvain Paris[‡] Adobe Research

Kayvon Fatahalian[§] Stanford University

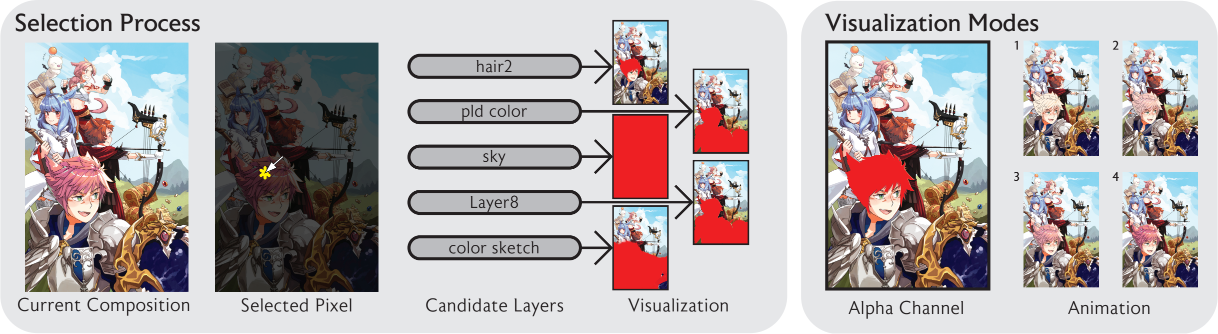

Figure 1: *Hover Visualization Interaction Method Overview.* To select a layer, the user first clicks on an area of the current composition. All of the layers that contribute to the selected pixel's color are displayed in a list of candidate layers. Hovering over the candidate layers triggers a visualization on the main canvas. There are three different visualization modes available (right). "Alpha Channel" displays the layer's alpha channel as a solid color. "Animation" displays a brief animation of the layer's opacity, rapidly fading it in and out, shown here as a series of four frames.

Abstract

In 2D digital art software, it is common to organize documents in to layers that are composited to create the final image, mimicking the traditional technique of creating an image by drawing on stacked transparent celluloid sheets. While intuitive, the layer stack suffers from problems of scale and organization. Documents with many layers are unwieldy to edit, finding a specific layer is akin to finding a needle in a haystack. This paper presents a *click and hover* interaction which visualizes the impact of a layer in the context of the full resolution composited image, providing an easier and faster way to identify layers in complex compositions. Through a user study, we find that users prefer to use hover visualizations in all cases, and that in compositions with many overlapping, semi-transparent layers, users are able to locate layers twice as fast with the click and hover interface.

Keywords: layers, hover, interaction method, layer visualization, image editing

Index Terms: Human-centered computing—Interaction paradigms; Human-centered computing—Interaction techniques; Human-centered computing—Graphical user interfaces; Applied computing—Media arts

1 Introduction

Layers are a fundamental part of image editing and composition. Layers allow digital artists to separate their composition into discrete parts, and achieve sophisticated effects with varying blend modes and adjustments. Digital artists interact with these layers on a daily basis, editing and arranging them to achieve their artistic vision. Layers in image editing software are typically selected and manipulated through a layers panel (Figure 2), which consists of a list of layer names and thumbnails. Some software eschews an explicit layer panel and instead opts to layer elements through simple "move forward/backward" ordering operations. Artist-created compositions have tens to hundreds of layers, many of which are semi-transparent, overlap, or have non-normal blend modes, complicating the seemingly simple task of finding the layer to edit. For example, the composition shown in Figure 4-c is an artist-created collage of 125 layers.

In documents such as Figure 4-c, it can be difficult to select a layer for editing. Without a layers panel, layers that are occluded by other layers are difficult or impossible to click on. Some software, such as Adobe Photoshop, solves this problem by providing a context menu containing all of the layers located under the cursor to allow selecting layers at deeper depths. With a layers panel, there are two ways to locate a layer: by name or through a manual search. Well-named layers provide clear indication of the layers contents (Figure 4-a), however many artists do not spend time naming their layers, viewing the process as tedious. This leads to many cases where layers are given non-meaningful names like "Layer 128" and "Layer 8," as seen in Figure 4-c. Names may also lose their meaning over time, as the artist has the ability to change the content of any layer at will.

The second method of identifying layers in the layers panel is through a manual search. This task is aided by thumbnail previews of the layers. The thumbnails provide limited information due to their size, and their portrayal of the layer pixels in isolation, as they are unable to display the effects of the layer's blend mode without any layers to blend. If the artist is unable to recognize the layer through the thumbnail, they are forced to rely on editing operations to understand what each layer does. The most common operation used is the visibility setting, which they toggle on and off in order to figure out what each layer does.

While the prior techniques are adequate for compositions with a small number of opaque objects, they are less suitable for finding semi-transparent layers in dense compositions, a common use case for intermediate and expert users. In this paper, we propose a set of surprisingly simple UI mechanisms for Photoshop-like applications that enables fast identification of where layers are and what they do.

[*] e-mail: eshimizu@cs.cmu.edu
[†] e-mail: matfishe@adobe.com
[‡] email: sparis@adobe.com
[§] e-mail: kayvonf@cs.stanford.edu

Our design is motivated by observing that in order to quickly identify a target layer, artists need to have the ability to quickly see the effect of their layers in the context of their composition. Our method comprises of a spatial filtering operation to select candidate layers that impact a pixel, and a hover operation on the resulting candidate layer list to reveal the effect of the layer on the composition at large. For this reason, we call our system a "click and hover" interface. We demonstrate the utility of the interface through a user study, finding that in compositions with many overlapping semi-transparent layers, the interface allows layers to be located twice as quickly.

2 Interface Design Principles

Layer selection is a universal task, and arises in many image manipulation situations. Through our own experiences with these tools, and in conversations with seven different artists, UX designers, illustrators, and students during the development of this interface, we believe that frustration with current layer selection interfaces arises from two primary factors. First, the process of finding a layer often requires the modification of the document itself because thumbnails do not provide enough information, and second, that this manual modification search becomes more tedious as the number of layers increases.

In our interviews, artists stated that they will try to use thumbnails to identify a layer, however the thumbnails sometimes do not provide enough information to locate a layer. In these cases, the default way to locate layers is to browse through the layer list and toggle the visibility of each layer in order to visualize the changes that it makes to the composition. They often have to repeat this for every layer that they want to locate, and it is difficult for them to remember where it is in the layer panel after they find it. Spatial selection methods help, but these mechanisms do not help when there are many visually-similar overlapping layers at the specified point. Naming layers is viewed as a tedious task by many artists, and even though they know layers should have useful names, they still sometimes choose not to name layers at the time of layer creation.

This problem is exacerbated as the number of layers in the composition grows. With hundreds of layers, it becomes more difficult for an artist to recall what each layer is and what it does in the composition. This problem is further complicated when the layer structure was created by a different artist (i.e. a collaborative workflow) or if the artist has simply not worked with the composition in a long time. One of the most time consuming tasks when working with such documents is figuring out how the document is structured, and then rearranging layers to fit current preferences. This process must be done with the manual modification search process for each layer that the artist needs to locate. Artists expressed interest in tools that would allow them to quickly browse through layers in this situation.

2.1 Design Goals

Based on conversations with artists, and our own observations, we believe that an interface for fast layer selection should follow the following design goals:

Inline Visualization Artists working in a visual medium need to see how layers affect the composition. They also need to be able to visualize these changes in the context of the composition itself, not in a separate panel or exploded view.

Transient Previews Previews of layer edits should not permanently modify the original composition.

Speed and Scalability Previews should be performed quickly enough to enable interactive layer browsing as the user hovers over a layer name. The interface should also be able to handle hundreds of layers.

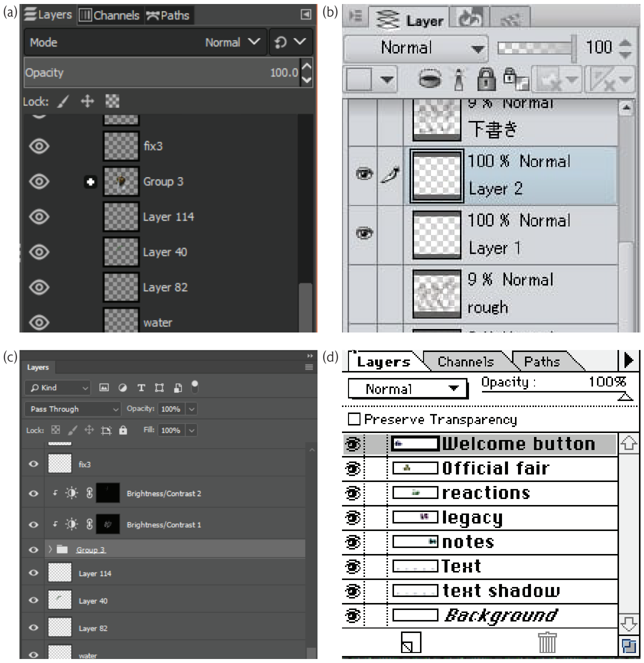

Figure 2: *Standard Layer Selection Interface.* Implementations of the standard "Layers" panel in consumer software. Pictured software: GIMP 2.10.6 [21] (a), Clip Studio Paint 1.7.8 [5] (b), Adobe Photoshop CC 2018 (19.1) [20] (c), and Adobe Photoshop 3.0 (d).

3 Related Work

The Layers Panel Layer selection in most image editing software [5, 7, 20, 21] is done through a layer panel (Figure 2). This interface displays layers ordered by depth, along with a user-assigned name, thumbnail, and controls for modifying layer depth, visibility, and blending settings. This interface paradigm has remained constant since layers were first added to commercial software packages in 1994 (Figure 2-d), ten years after Porter and Duff introduced alpha compositing in 1984 [16]. Some layers panels in commercial software, such as Microsoft Powerpoint [15], display control handles on a bounding box containing the selected object. While this works well for vector shapes with distinct outlines, it can be difficult to properly scale bounding boxes to indicate the proper extents of bitmap layers. Our method is designed to enhance the current layers panel interface paradigm.

Spatial Selection Spatial selection can be used to filter out layers that do not affect a point or region of interest. This type of selection feature has been present in 2D interfaces for raster and vector graphics since 2010 (Photoshop 12.0 [1]) at the latest. This type of selection method is equivalent to raycasting in 3D interfaces, and can be thought of as a 2D ray selection tool (click) with a manual disambiguation metchanism (hover) [2, 19].

Even with spatial selection, the problem of disambiguating layers under the cursor remains. LayerFish [22] presents a touch-based interaction that combines spatial selection with a fisheye menu containing layer thumbnails. The fisheye menu provides thumbnails of the layers with layers that are further apart in depth appearing smaller (creating a fisheye effect). LayerFish relies on thumbnails and spatial transformations, and while the fisheye menu resolves the problem of working with large numbers of layers seen in [17], it is still subject to the same difficulties found in performing layer identification with thumbnails alone. The system was shown to be effective for quickly browsing through a large number of layers in a list, however it is unclear how much benefit comes from spatial selection versus the fisheye display mechanism.

Tumbler and Splatter [17] allows users to access opaque, occluded objects in a composition by applying spatial transformations in order to visualize the depth ordering of the layers in the scene. This interface "explodes" the document in order to show the depth relationships between the layers. The Tumbler and Splatter interaction method applies spatial transformations to the layers, removing them from their original context. With this method, finding layers that have a distinct shape or color among other opaque layers is easy, but finding, for instance, a semi-transparent layer with a non-"normal" blend mode (i.e. not a Porter-Duff OVER [16] operation) is difficult, as the transformed layer thumbnail does not represent what the layer looks like in context. This type of interface also has difficulty scaling as the number of layers increases.

Transparency In windowed environments, multi-blending [3] makes content occluded by opaque windows visible by changing the blending mode of the window. Along similar lines, content-aware free-space transparency [14] utilizes unused window backgrounds to display occluded content. We are unable to use multi-blending techniques in an image editing context, as artists use blend modes to achieve specific artistic effects, and we must respect those choices. In the case of free-space transparency, it is unclear what parts of the artist's composition count as "free-space," and we cannot reliably use this technique because not all of the layers are fully opaque.

In 3D selection tools, transparency-based techniques are commonly used to assist with selecting occluded objects. Hinckley et al. [12] surveys a number such selection techniques, including utilizing a semi-transparent plane to provide occlusion clues, making foreground objects transparent, and dynamically adjusting object transparency during an editing task [8, 9]. The problem with using these techniques in a 2D image editing context is that the layers are not occluded, but instead contribute to effects that are already visible in the document. These effects arise from a combination of layer structure and artistic layer blending modes, and changing transparency alone is often not sufficient to locate layers. We have implemented a transparency animation visualization method (Section 4.2.2) in the click and hover interface to investigate the effectiveness of transparency-based selection assistance in the 2D image editing context.

In-Context Visualiations Toolglass and Magic Lenses [4] present a method of previewing changes by utilizing a floating window (lens) which visualizes filters that could be applied to a visual element. These filters can be anything from transparency to complex spatial warps of the image content. The click and hover system can be viewed as a transient application of a lens that operates on the hovered layer and displays its effect across the entire canvas. Magic Lenses and the click and hover interface are examples of interference minimizing interfaces, as presented by Harrison et al. [11], that are designed to allow users to stay aware of their composition's state while focusing on a selection or manipulation task.

Dynamic Selections and Input Mechanisms Other methods for assisting layer selection include automatically selecting objects based on selection history [18]. This method selects groups of objects from a single selected item, using the selection history as a prior that informs how the selection expands. This method is designed for vector objects, and could be extended to bitmap applications, however it does not solve the problem of making the initial selection. Grossman et al. [10] present a selection system for virtual pen and ink systems that combines spatial selection with automatic selection grouping by showing possible selection sets that a single ink stroke could belong to. The click and hover interface can work with these types of automatic selection assistance by providing better tools for making the initial selection.

In touch-based environments, there have been efforts to utilize the "transparent sheets" metaphor of layers to provide intuitive touch controls for layering. Davidson and Han [6] present a set of pressure-sensitive touch gestures that support layering tasks on touch interfaces. Hinckley et al. [13] use observations of how users handle physical pen and paper to inform a set of interaction techniques with a virtual pen and paper. The click and hover interface is not designed to support layer manipulation; instead, the interface supports selection, which is the first step towards manipulating a layer. Supporting fast selection should benefit these types of existing methods by reducing the time spent locating a layer instead of manipulating it.

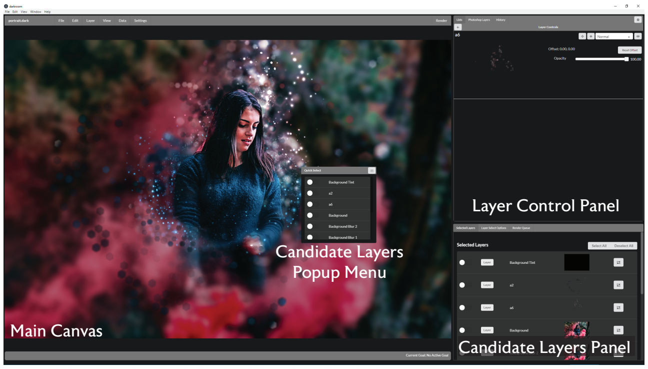

Figure 3: *Interface Overview* Our interface is divided into three main sections. The Main Canvas displays the current rendering of the composition. The Candidate Layers Panel displays a list of layers that are under the pixel selected by a user. The Layer Control Panel displays controls for manipulating the layer selected from the candidate layers panel. Users are able to view the candidate layers panel as a pop-up menu by right clicking on the canvas.

4 System Description

To resolve the difficulties associated with finding layers in complex image compositions, we propose a two-step *click and hover* process. We implement this process in our interface, shown in Figure 3. Users first perform a spatial filtering operation by *clicking* a point on the canvas where the layer they want exists. Multiple layers may exist at this location, so our interface provides a set of layer visualization tools that allow the user to perform a quick visual search to locate the specific layer they are interested in. The layer visualization tools are activated by *hovering* over the layer name in a separate panel after performing the spatial filter. Once found, our interface allows users to adjust the selected layer. In this section, we describe how the click and hover interface works.

4.1 Click to Localize

To begin the selection process, the user clicks on a region of the image, and the system computes a set of candidate layers that have a non-zero contribution at the clicked pixel. After determining which layers affect at the selected pixel, the layers are displayed in the *candidate layers panel* in the bottom right side of our interface (Figure 3-bottom right). Adobe Photoshop exposes the spatial selection capability through a context menu instead of a panel, so in order to maintain parity with the current state-of-the-art, we also provide the ability to view the candidate layer list in a popup menu (Figure 3-center) through a right click operation. The popup menu does not contain thumbnails, but the list of layers shown is identical to the candidate layers panel. This allows users to explore the image spatially, not through the layer hierarchy. This is particularly useful for situations where the user does not know how the layers are ordered, but does know where the elements they would like to modify exist.

4.2 Hover to Visualize

In a composited image, multiple layers can contribute to the final color of a pixel in the image. Therefore, in order to easily identify a specific layer, there should to be a mechanism to view the contribution of the layer that affects a pixel in the context of the original document. In the candidate layers panel, we use a hover mechanism that visualizes the hovered layer on the main canvas. Compared to thumbnail views, on-canvas visualizations eliminate the guesswork the artist does to map the transformed thumbnail to a location on the full size canvas, enabling easy visualization of small elements that are hard to see in thumbnails, and providing the opportunity to provide more information about how the layer affects the overall look of the composition. In this way, the user can determine which layers do what with a few waves of the mouse after only one click.

The hover mechanism provides a way to display information about a layer in the same context as the composition. There are many possible ways to visualize the effect of a layer. We propose two such visualization methods inspired by current practices with image editing tools (Figure 1-right).

4.2.1 Alpha Channel

One of the primary aspects of a layer that artists look at is shape. The alpha channel visualization displays the alpha channel of the layer, modulated by the current opacity of the layer and the alpha value of the layer pixels, as a solid color on top of the current composition. The resulting solid color layer is alpha blended on top of the current composition. Layers that are more transparent will have a less intense color, and layers that are hidden or do not contribute to the composition will not show up.

This visualization provides instant feedback regarding the shape of the layer on the canvas itself. For occluded objects, the visualization exposes the shape of the occluded part of the layer, as it is presented on top of the final composition. This type of visualization excels when the layers in question have unique outlines and do not cover the entire canvas, but it may lead to some confusion if the hovered layer is occluded, as this visualization will display the layer on top of the overall composition.

The default color for the alpha channel visualization is pure red, the default alpha mask color in Photoshop. This color can be customized by the user if it conflicts with the content of the image composition.

4.2.2 Opacity Animation

The opacity animation visualization makes the hovered layer visible, and then animates its opacity from 0% to 100% over a short duration. The layer's blend mode is unchanged, as is its position in the image composition. This creates a flicker effect that reveals the current effect that the transparency of the layer has on the composition. This visualization method imitates the artist's current layer search methods, specifically that they toggle the visibility of a layer in order to determine what it does.

The animation visualization is designed to provide feedback about how the current layer settings affect the overall composition. This can be helpful for identifying layers that utilize non-normal blend modes, full canvas layers, and layers that are heavily occluded. This visualization does not provide instant feedback, as the animation does take a short time to run.

Our implementation of the opacity animation runs a 25 frame loop at 60 frames per second, animating for 10 frames, and holding the layer at full opacity for 15. This effectively flickers the layer 2.4 times per second. These settings are customizable by the user. In order to run the interface at interactive rates, our implementation of the animation downsamples the image to approximately 25% of the original resolution. Faster renderers should be able to run at higher resolutions.

5 Evaluation

We conducted a within-subjects user study to investigate how much faster the click and hover interface allowed users to locate layers in complex image compositions, relative to techniques present in current image editing software (e.g. Adobe Photoshop). Qualitative data about the perception of the effectiveness of the visualization types was also collected.

5.1 Methodology

5.1.1 Participants

15 participants age 20-45 were recruited (5 male, 10 female). Participation was voluntary, and no compensation was given. Participants of all skill levels were allowed to participate in the study. Most participants had some experience with software such as Adobe Photoshop, and about 20% of participants described themselves as experts. The participants in this study did not participate in the preliminary interviews described in Section 2.

5.1.2 Interface Configuration

Participants used three different configurations of the hover visualization interface: baseline, alpha, and animation.

Baseline The baseline interface configuration allows users to use spatial selection to filter out layers that do not affect the pixel under the cursor. No hover visualizations appeared when hovering over the layer names. This is representative of the current feature set in Photoshop, however we note that spatial selection is not a universally implemented technique in image editing software, and that this baseline is slightly stronger than a comparison against the layer panel alone.

Alpha The alpha interface configuration shows the alpha channel visualization described in Section 4.2.1 when a layer name is hovered over in addition to allowing spatial selection. The color of the alpha channel visualization was red, and users were not allowed to change the color.

Animation The animation interface configuration shows the opacity animation visualization described in Section 4.2.2 when a layer name is hovered over in addition to allowing spatial selection. The animation runtime is as described in Section 4.2.2, and users were not allowed to change the timing of the animation.

Participants were not allowed to change the visualization type while using an interface configuration. Participants had the option of using the candidate layers panel or the candidate layers popup menu (Figure 3) to display results from the spatial selection. Spatial selection results are ordered top to bottom by layer depth (standard layer panel ordering).

5.1.3 Tasks

In this study, users were asked to find a highlighted layer in an image composition as shown in Figure 4. The target images shown in Figure 4 are created with an "overwrite color" operation, which changes the color of all the layer pixels to a single RGB color, while leaving the alpha channel alone. Users indicated their layer selection by performing the overwrite color operation with a non-white color (the color did not have to match the highlight color in Figure 4). If the user modified the correct layer, the task ended. If not, the task continued and the user would select another layer. The task did not end until the user selected the correct layer. Four compositions were chosen to investigate different properties of the hover interface:

Illustration (Figure 4-a) 116 Layers. This image consists of well-named layers with sharply defined outlines. Due to appropriate

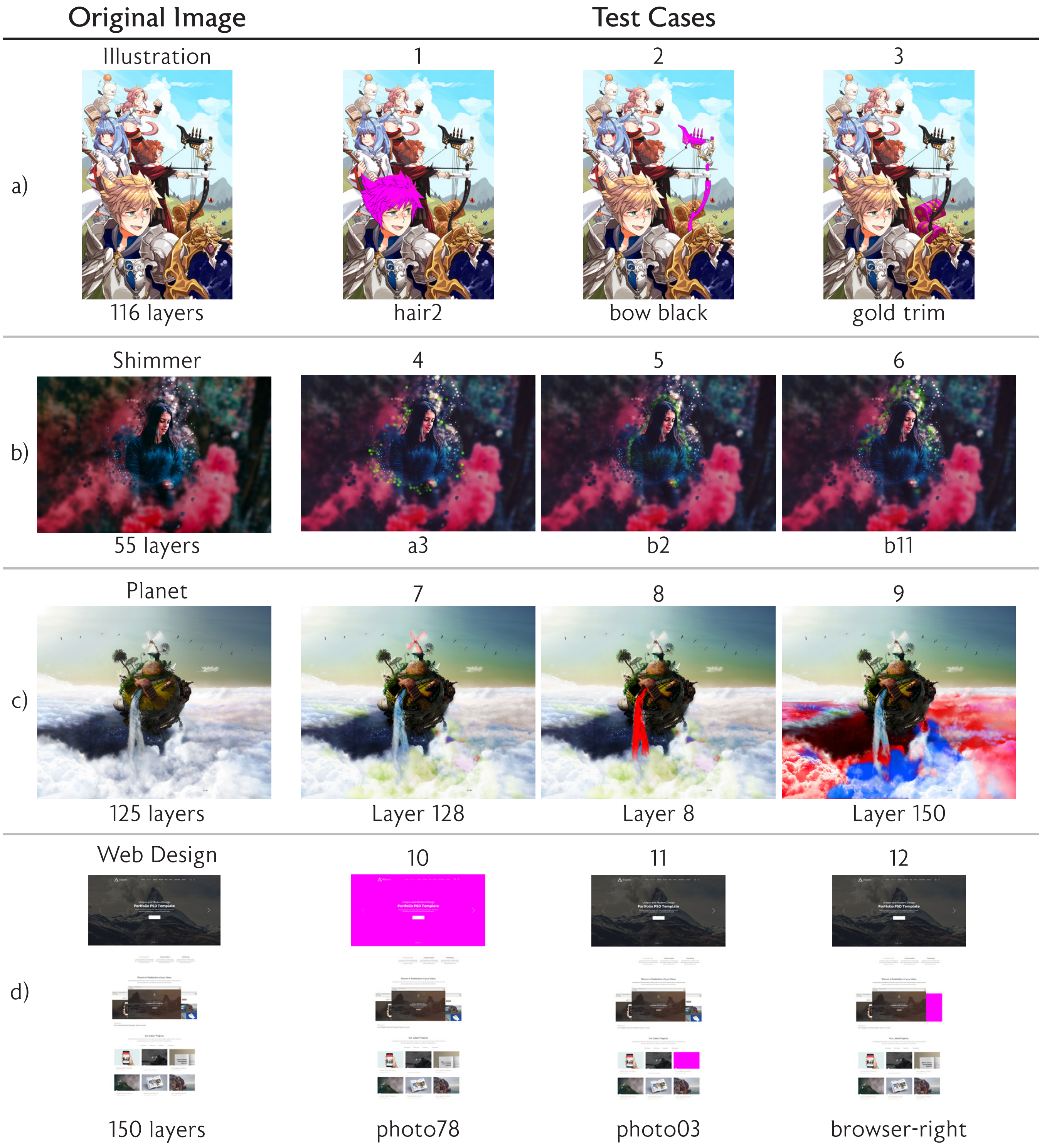

Figure 4: *User Study Tasks.* Each task asked the user to find the highlighted layer by performing and overwrite color operation, replacing the color of the layer pixels with a solid color while leaving the alpha channel unchanged. Original compositions are shown on the left with the composition name and total number of non-adjustment layers. The task number is listed above the test cases, and the name of the target layer is listed below the test cases. The test case images are shown as they were presented in the user study. In these images, the target layer is highlighted in pink (a, d), green (b), and red (c).

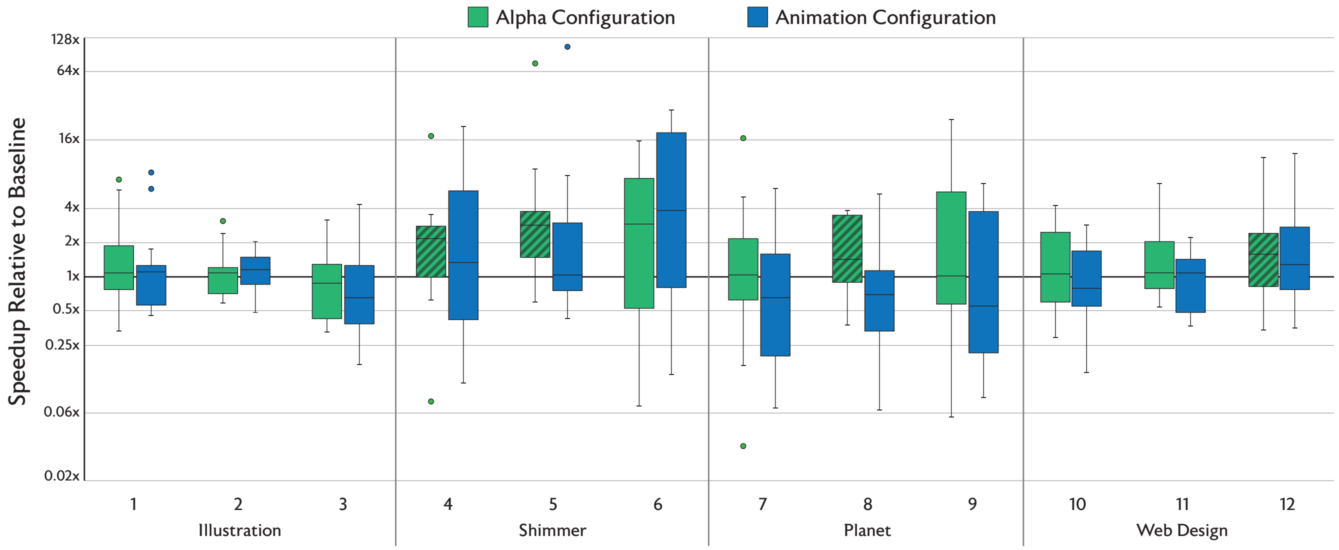

Figure 5: *Median Task Speedup Relative to Baseline*. Box and whisker plot of the per-task distributions of the task speedup relative to the baseline. Outliers are shown as single points. The y-axis is plotted on a $\log_2$ scale. The speedup is the time it takes for a user to find the correct layer with the listed interface configuration relative to the time it took for them with the baseline configuration. Results that are significant at the $p < 0.05$ (Table 1) level according to the Wilcoxon rank sum test against the constant baseline distribution (median 1) are indicated with a diagonal striped background. The alpha configuration generally meets or exceeds performance relative to the baseline on all tasks, while the animation configuration appears to help in some circumstances, it does not consistently improve performance.

Table 1: *Wilcoxon Rank Sum Test p-values*. The speedup distributions for each task (Figure 5) were compared against a constant distribution with median 1 (no speedup) using the Wilcoxon rank sum test. Distributions with a significantly different median (at $p < 0.05$) are bolded.

Composition	Task	Alpha	Animation
Illustration	1	0.33	0.33
	2	0.33	0.10
	3	0.33	0.33
Shimmer	4	**0.007**	0.33
	5	< **0.001**	0.76
	6	0.10	0.33
Planet	7	0.76	0.10
	8	**0.02**	0.10
	9	0.33	0.33
Web Design	10	0.33	0.76
	11	0.76	0.76
	12	**0.02**	0.76

naming and organization of layers, we do not expect the hover visualizations to provide much additional benefit to the users, and performance should be on par with the baseline interface.

Shimmer (Figure 4-b) 55 Layers. This image was created by a Photoshop Action, an automated script that applies an effect effect to an input image. The layers created by the script are not semantically named (most consist of a single letter and number, e.g. "a2") and can be difficult to identify from thumbnails alone. This represents a common class of compositions that have a large number of semi-transparent overlapping layers that are difficult to identify with the layers panel alone. We expect the hover visualizations to perform better than the baseline here.

Planet (Figure 4-c) 125 Layers. This artist-created image utilizes a number of non-normal blending modes and unnamed layers to create the composition. Some layers appear identical and are layered on top of each other (Task 7), some layers are part of an animated sequence, where only one layer is intended to be visible at a time (Task 8), and some drastically change the appearance of the composition but only when their color is changed (Task 9). These are all difficult cases for the hover visualizations, and we investigate to what extent they help or hinder users in these tasks.

Web Design (Figure 4-d) 150 Layers. This composition has the most layers of any composition in our tasks, however, at any particular pixel there are at most four layers. This sort of composition is difficult to navigate with the layers panel alone, but should prove to be easy with spatial selection. As with the Illustration image, we expect this task to be easily completed with all three interface configurations, and include it here to demonstrate the strength of the spatial selection baseline.

Each composition has three tasks, and the tasks are repeated for each interface configuration, leading to a total of 36 tasks. The tasks are performed in sets of 12, where the order of the compositions is randomized, and then the order of the tasks associated with each composition is also randomized. The per-participant task order between interfaces is the same, ensuring that tasks for a particular composition are not inadvertently placed consecutively. The interface order is counterbalanced in order to account for learning effects. Upon completing all 36 tasks, participants answered a short survey. The study took 40-60 minutes to complete.

5.1.4 Metrics

Timing data is collected from interaction logs recorded by the interface. The timer starts when the user is presented with the target layer image. From these logs, we extract timing information including the time to selecting the right layer. We also record the number of times a layer was selected, and the number of identification errors made.

We collect qualitative data from an exit survey. The text of the exit survey questions is shown on the left side of Figure 6.

Table 2: *Wall Clock Task Times and Total Errors*. Median and average completion times (with σ) for each task in seconds. Lower is better. Errors are defined as the number of times a participant incorrectly identified the layer in the task before identifying the correct layer, and the total number of errors made by participants on each task is reported here. Best values for each metric in each task are bolded.

		Baseline			Alpha			Animation		
Composition	Task	Median	Avg (σ)	Err	Median	Avg (σ)	Err	Median	Avg (σ)	Err
Illustration	1	5.4	9.7 (10.0)	1	**5.0**	**5.0 (3.4)**	**0**	5.5	6.7 (3.3)	**0**
	2	6.4	7.2 (3.2)	0	6.0	**6.7 (2.9)**	0	**5.6**	6.8 (4.3)	0
	3	**7.2**	**7.9 (4.1)**	**0**	7.5	10.6 (7.3)	**0**	11.1	12.1 (7.6)	1
Shimmer	4	34.9	57.0 (57.4)	9	20.5	**37.0 (54.7)**	4	**19.6**	37.2 (40.3)	**0**
	5	35.4	107.3 (169.1)	9	**12.6**	**23.3 (20.7)**	**2**	29.5	38.7 (25.2)	5
	6	54.6	103.0 (113.3)	12	18.0	40.5 (53.7)	1	**14.9**	**26.7 (29.1)**	**1**
Planet	7	20.6	30.5 (32.9)	**4**	**13.8**	**28.5 (28.5)**	9	27.9	44.5 (39.8)	11
	8	32.7	29.8 (18.3)	1	**12.5**	**16.6 (8.9)**	**0**	34.1	46.7 (35.0)	2
	9	43.0	56.1 (61.2)	8	**28.8**	**33.3 (29.1)**	**6**	47.6	66.9 (60.2)	9
Web Design	10	5.4	6.5 (3.6)	**0**	**5.0**	**5.7 (3.9)**	**0**	6.2	7.7 (7.2)	1
	11	5.6	5.7 (2.7)	0	**4.2**	**4.5 (2.0)**	0	5.8	6.3 (2.7)	0
	12	8.3	16.1 (21.3)	3	**5.5**	**6.4 (2.5)**	**0**	7.2	7.6 (3.9)	**0**

These questions were answered on a 5-point scale. Additionally, participants self-reported their experience level with image editing or similar design tools. Participants were also given the opportunity to give open feedback regarding what they liked and disliked about the hover visualizations at the end of the survey.

5.2 Analysis

Quantitative results from the user study are presented in Figure 5 and Table 2. Qualitative results are presented in Figure 6. Performance on the Shimmer tasks improved by a median factor of 2x with the hover interfaces, performance on the simpler tasks of Illustration and Web Design improved slightly, and performance on the Planet tasks reveal the limits of the presented visualization methods. In particular, we find that the alpha configuration strictly improves on the baseline configuration, demonstrating significantly improved performance on the difficult Shimmer tasks, and performing no worse than the baseline on all other cases. Qualitatively, users enjoyed using the hover interface more than the baseline, praising its ability to provide instant feedback without drawing attention away from the main canvas. The alpha configuration was perceived to be easier and faster to use than the animation and baseline configurations by all users (Figure 6).

5.2.1 Quantitative Metrics

In order to normalize time metrics between participants, we compute the improvement factor between the baseline and the alpha and animation configurations by using the "time to selecting the correct layer" data recorded in the study. The medians of the improvement factors for each task are shown in Figure 5. We report median speedup instead of average speedup due to extreme outliers in some of the tasks. Median and average task completion time, along with standard deviations and total errors, are reported in Table 2 to give a sense of the time scale for the tasks.

Where Click and Hover Helps Median performance in tasks involving the Shimmer composition with the alpha configuration improved by 2.1-2.8x ($p < 0.05$ for tasks 4 and 5, $p < 0.11$ for task 6). These tasks involved layers with a large number of separate scattered elements on essentially unnamed layers. Under these circumstances, the alpha visualization let users quickly navigate through the layer stack, and provided more information about the layers location in the canvas than the available thumbnails. In contrast, the animation configuration saw some improvement, but not to the same significance of the alpha configuration.

In these tasks, the hover visualizations also made participants more precise, reducing the number of errors made from a total of 30 in the baseline, to 7 and 6 for the alpha and animation configurations respectively. Errors are defined as the number of times the wrong layers were modified with an overwrite color operation before modifying the correct layer. In other words, the participants did not need to modify the composition as much with the alpha and animation configurations as they did with the baseline configuration, satisfying the transient previews design goal of the click and hover interface. Hover visualizations also reduced the wall clock completion time variance in tasks 4-6, suggesting that all participants had an easier time completing the task with visualization configurations. Most participants favored the alpha channel visualization, viewing the animation as slower (Figure 6) and not providing enough feedback about the boundaries of the hovered layer. Participants were more precise with both hover visualizations configurations compared to the baseline (Table 2).

Hover Visualiations Do Not Hinder Simple Cases In cases where the layers are well named or sparse, the hover visualizations do not impede the layer selection process, and should be safe to add to existing image editing software without adversely affecting existing processes. We observe that there is no significant difference in speedups in either the alpha or animation configurations for tasks involving the Illustration and Web Design compositions. Task performance on Illustration is best explained by noting that all of the target layers had semantically meaningful names (i.e. "Hair2" corresponds to a layer containing a character's hair) and used mostly standard blend modes. Under these circumstances, users could simply use the layer names without relying on the hover visualizations.

On the Web Design tasks, spatial selection provided participants with a list of at most three layers when looking for the highlighted layer. The time it takes to manually inspect those layers is negligible, and performance improvements were minor. We note that task 12 had a median improvement of 1.5x when using the alpha configuration, which was significant ($p < 0.05$). This task involved finding an occluded layer, and some users found the thumbnail to be unhelpful in this circumstance.

Where Hover Visualizations Struggle Median relative performance in the tasks involving the Planet composition generally went down while using the animation configuration (except in task 8), and were approximately equivalent to baseline while using the alpha configuration. Each task in this composition has some unique challenges highlighting the limits of the visualizations presented here.

In task 7, there are two identical layers located at the highlighted location ("Layer 128" and "Layer 128 Copy"). Both layers were depth-adjacent and appeared next to each other in the candidate layers list. Layer 128 is above Layer 128 Copy and was placed before Layer 128 Copy in the candidate layers list. Neither visualization was quite able to help participants identify which of the two similar layers is highlighted in the target image, as the only difference is a slight transparency differences. This situation had fewer errors using the baseline configuration, likely because participants worked from the top of the candidate layer list to the bottom, instead of immediately exploring the other options as they did with the hover visualizations.

Task 8's target layer is co-located with a large number of similar layers. There is only one layer (the target, Layer 8) that is currently visible, all other similar layers are turned off (opacity 0%). In this instance, the alpha visualization will only activate on the one visible layer, making this particular task almost trivial and providing a median 1.4x speedup. The animation visualization did not provide any assistance on this task, as it does not respect the current visibility settings of the layers. Because the animation provided no assistance, the task in this configuration was reduced to the baseline.

Task 9 asked participants to identify a layer using a non-standard blend mode. The specifics of this task caused problems for the visualizations due to the layer only revealing its color effects when a non-white color adjustment is applied. In this case, the animation configuration failed to provide useful information by just adjusting the transparency, and the alpha configuration had trouble because it was difficult for participants to tell where the highlighted layer applied the relevant effect.

Learning Effects There may be a learning effect within participants, as they may be able to remember the layer names for the compositions in subsequent trials. To evaluate the strength of this effect, we ran an 1-way ANOVA over the per-task ratio distributions for the alpha and animation configurations, using interface order as the grouping variable. If there is a strong learning effect, we would expect the means of the ratios grouped by order to be different. We found no significant effect from order on the alpha configuration ($p < 0.05$), but did find a a significant effect on tasks 6, 8, and 10 ($p < 0.05$) with the animation configuration.

The learning effect on task 8 is likely due to the animation visualization providing no additional information for that task. In later repetitions of the tasks, participants generally ignored the animation visualization and repeated the process they had already done in the baseline configuration, leading to faster task completion times. The learning effect on task 6 is possibly due to the difficulty in seeing the layer's effect, as it is towards the bottom of the layer hierarchy. Users may have remembered the difficulty of this layer, and relied on that instead of the animation visualization. Task 10's learning effect is difficult to explain. The interface choice did not significantly affect the performance on this task, and yet there is a statistically significant ordering effect.

5.2.2 Qualitative Metrics

Responses to the survey questions are presented in Figure 6. Overall, participants strongly preferred using the hover configurations over the baseline interface, stating that it was both easier and faster to find layers using the alpha channel visualization. One participant explained their responses by saying that "the Alpha channel required very little cognitive effort. [The visualizations] would be very useful on compositions with a lot of layers with intricate effects." (participant 5)

The color of the alpha channel visualization was fixed to pure red in the alpha configuration. Almost all participants found this color choice to be acceptable for the four compositions in the study, although one participant remarked that they "would like to be able to change the highlight color – some colors are more appropriate to some tasks than others." (participant 9)

The ability to obtain instant feedback without needing to look at different parts of the interface to determine layer location and effect is viewed as a positive by many participants. Participants specifically praised the click and hover system for providing "instant feedback about the layer [without] having to look at the little [thumbnail]" (participant 8) and that the alpha visualization specifically is "is easy to flip through" (participant 9). This is a good indication that the click and hover system accomplishes the speed and scalability design goal presented in Section 2.

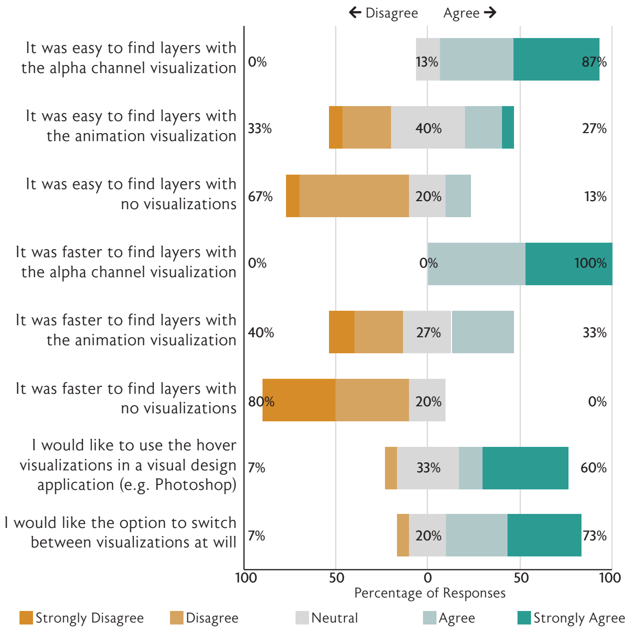

Figure 6: *Survey Responses*. Likert plot of responses to the survey questions. Participants overwhelmingly favored the alpha channel visualization, and most expressed interest in bringing the hover visualizations (in some form) to commercial editing software.

Participants viewed the alpha channel as faster than the animation visualization, and given a choice, 80% of participants expressed interest in only using the alpha channel visualization. The speed was one of the primary concerns that participants had with the animation. Many noted that the animation felt too slow and wanted it sped up. Speeding up the animation further may create a strobe effect that would be unsuitable for photosensitive users, so any implementation of this type of visualization should be be mindful of that effect.

Participants expressed some interest in putting hover visualizations in existing commercial image editing software. If it were to be implemented, participants noted that they would like the ability to control which visualization get activated, as they found that the visualization could be jarring. We recommend that implementations of the visualizations in image editing software provide the adequate controls to fit visualizations to user preferences.

5.2.3 Limitations

Our evaluation focused on two simple visualization techniques. These methods (alpha and opacity animation) performed well relative to the keyboard and mouse-based baseline found in popular contemporary image editing software. The alpha visualization was found to be most effective, yet still struggled with layers that use non-"normal" blend modes with unclear boundaries, as shown in Task 9 (Figure 4). Improved visualization methods may take inspiration from Magic Lenses [4] in order to provide visualizations that are more relevant to the artist's task.

6 Discussion

In this paper, we introduced the click and hover interface and evaluated two visualizations that can be used with the system. Through our evaluation of the visualizations, we find that the click and hover system effectively augments the existing layer manipulation interfaces in commercial software, providing additional information about the location and effect of layers that names alone cannot provide. Selection is the first step for many interactive workflows, and this simple interaction method provides a foundation for improving tools that allow artists and designers to edit complex documents, letting them focus less on fighting the interface, and more on their art.

We are interested in continuing to explore new hover visualization techniques. Our current hover visualizations provide insight into where the layers are in the composition, but an interesting area of future work is to explore hover visualization techniques that better suggest what could be done with the layer. The ability to provide speculative visualizations, showing what the user could do instead of what is currently being done, may resolve the difficulties encountered in the our study's Planet tasks.

The click and hover interface is designed for, and was compared against, traditional mouse and keyboard interfaces, and can be extended to touch interfaces. Instead of a click and hover action, a touch interface could utilize a press and slide interaction technique. In this interface, the layer visualizations would be activated by the user pressing and holding on a location on the canvas (click), and could activate visualizations by sliding up and down to scroll through the candidate layers list (hover). The visualization techniques remain the same, but the interactions that activate them change to fit the properties of touch-based input interfaces.

Source Code

The source code for the prototype hover visualization interface can be found at github.com/ebshimizu/sliders.

Acknowledgements

The authors would like to thank Daichi Ito and Joy Kim from Adobe for providing image compositions for this project, and Jim McCann for paper writing and editing assistance. This research was supported by the National Science Foundation (IIS-1253530) and by gifts from Adobe.

References

[1] Move tool moves wrong layer... — adobe community. `https://forums.adobe.com/thread/998691`. Accessed: 12-15-2018.

[2] F. Argelaguet and C. Andujar. A survey of 3d object selection techniques for virtual environments. *Computers and Graphics*, 37(3):121 – 136, 2013. doi: 10.1016/j.cag.2012.12.003

[3] P. Baudisch and C. Gutwin. Multiblending: Displaying overlapping windows simultaneously without the drawbacks of alpha blending. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, pp. 367–374. ACM, New York, NY, USA, 2004. doi: 10.1145/985692.985739

[4] E. A. Bier, M. C. Stone, K. Pier, W. Buxton, and T. D. DeRose. Toolglass and magic lenses: The see-through interface. In *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '93, pp. 73–80. ACM, New York, NY, USA, 1993. doi: 10.1145/166117.166126

[5] CELSYS. Clip studio paint, 2018.

[6] P. L. Davidson and J. Y. Han. Extending 2d object arrangement with pressure-sensitive layering cues. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology*, UIST '08, pp. 87–90. ACM, New York, NY, USA, 2008. doi: 10.1145/1449715.1449730

[7] S. S. Development. Painttool sai, 2016.

[8] N. Elmqvist, U. Assarsson, and P. Tsigas. Employing dynamic transparency for 3d occlusion management: Design issues and evaluation. In *Proceedings of the 11th IFIP TC 13 International Conference on Human-computer Interaction*, INTERACT'07, pp. 532–545. Springer-Verlag, Berlin, Heidelberg, 2007.

[9] C.-W. Fu, J. Xia, and Y. He. Layerpaint: A multi-layer interactive 3d painting interface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pp. 811–820. ACM, New York, NY, USA, 2010. doi: 10.1145/1753326.1753445

[10] T. Grossman, P. Baudisch, and K. Hinckley. Handle flags: Efficient and flexible selections for inking applications. In *Proceedings of Graphics Interface 2009*, GI '09, pp. 167–174. Canadian Information Processing Society, Toronto, Ont., Canada, Canada, 2009.

[11] B. L. Harrison, H. Ishii, K. J. Vicente, and W. A. S. Buxton. Transparent layered user interfaces: An evaluation of a display design to enhance focused and divided attention. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '95, pp. 317–324. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, 1995. doi: 10.1145/223904.223945

[12] K. Hinckley, R. Pausch, J. C. Goble, and N. F. Kassell. A survey of design issues in spatial input. In *Proceedings of the 7th Annual ACM Symposium on User Interface Software and Technology*, UIST '94, pp. 213–222. ACM, New York, NY, USA, 1994. doi: 10.1145/192426.192501

[13] K. Hinckley, K. Yatani, M. Pahud, N. Coddington, J. Rodenhouse, A. Wilson, H. Benko, and B. Buxton. Pen + touch = new tools. In *Proceedings of the 23Nd Annual ACM Symposium on User Interface Software and Technology*, UIST '10, pp. 27–36. ACM, New York, NY, USA, 2010. doi: 10.1145/1866029.1866036

[14] E. W. Ishak and S. K. Feiner. Interacting with hidden content using content-aware free-space transparency. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*, UIST '04, pp. 189–192. ACM, New York, NY, USA, 2004. doi: 10.1145/1029632.1029666

[15] Microsoft. Microsoft powerpoint, 2019.

[16] T. Porter and T. Duff. Compositing digital images. *SIGGRAPH Comput. Graph.*, 18(3):253–259, Jan. 1984. doi: 10.1145/964965.808606

[17] G. Ramos, G. Robertson, M. Czerwinski, D. Tan, P. Baudisch, K. Hinckley, and M. Agrawala. Tumble! splat! helping users access and manipulate occluded content in 2d drawings. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, AVI '06, pp. 428–435. ACM, New York, NY, USA, 2006. doi: 10.1145/1133265.1133351

[18] S. L. Su, S. Paris, and F. Durand. Quickselect: History-based selection expansion. In *Proceedings of Graphics Interface 2009*, GI '09, pp. 215–221. Canadian Information Processing Society, Toronto, Ont., Canada, Canada, 2009.

[19] J. Sun, W. Stuerzlinger, and D. Shuralyov. Shift-sliding and depth-pop for 3d positioning. In *Proceedings of the 2016 Symposium on Spatial User Interaction*, SUI '16, pp. 69–78. ACM, New York, NY, USA, 2016. doi: 10.1145/2983310.2985748

[20] A. Systems. Adobe photoshop cc 2018, 2018.

[21] T. G. Team. Gnu image manipulation program, 2018.

[22] A. M. Webb, A. Kerne, Z. Brown, J.-H. Kim, and E. Kellogg. Layerfish: Bimanual layering with a fisheye in-place. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces*, ISS '16, pp. 189–198. ACM, New York, NY, USA, 2016. doi: 10.1145/2992154.2992171