

Tangible Viewports: Getting Out of Flatland in Desktop Environments

Renaud Gervais

Inria, Bordeaux, France

renaud.gervais@inria.fr

Joan Sol Roo

Inria, Bordeaux, France

joan-sol.roo@inria.fr

Martin Hatchet

Inria, Bordeaux, France

martin.hachet@inria.fr

ABSTRACT

Spatial augmented reality and tangible interaction enrich the standard computer I/O space. Systems based on such modalities offer new user experiences and open up interesting perspectives in various fields. On the other hand, such systems tend to live outside the standard desktop paradigm and, as a consequence, they do not benefit from the richness and versatility of desktop environments. In this work, we propose to join together physical visualization and tangible interaction within a standard desktop environment. We introduce the concept of Tangible Viewport, an on-screen window that creates a dynamic link between augmented objects and computer screens, allowing a screen-based cursor to move onto the object in a seamless manner. We describe an implementation of this concept and explore the interaction space around it. A preliminary evaluation shows the metaphor is transparent to the users while providing the benefits of tangibility.

ACM Classification Keywords

H.5.1 Multimedia Information Systems: Artificial, augmented, and virtual realities; H.5.2 User Interfaces: Interaction styles

Author Keywords

Tangible Interaction; Spatial Augmented Reality

INTRODUCTION

Look at your work space right now. There is a high probability that the space is divided into two different areas: one for working digitally (computer) and one for working physically (pen and paper, books, building materials). This dichotomy has been present in our work environments for a long time, and a lot of effort of the TEI community has been directed towards a digitally enriched physical space. Compared to the traditional mouse-based paradigm of computers, tangible interaction [12] has been shown to provide richer interaction experiences that are especially well suited for collaboration, situatedness and tangible thinking [28]. On the other hand, even when tangibility hold great promises for interaction, its use in real-world contexts remains rare, while we still use standard computers

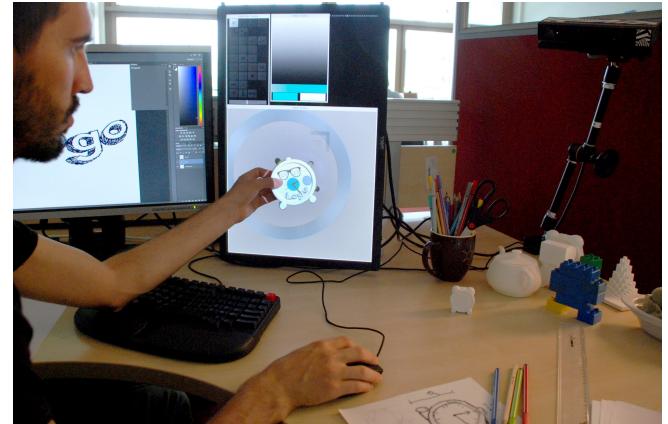


Figure 1. A workspace that allows the user to work both digitally (left) and physically (right). A real object can be placed in front of the screen to use it as a canvas for digital applications.

for the majority of our daily tasks involving digital information. The desktop computer is still a relevant tool to work with digital and physical matter, but we also think that its place on our desks should be rethought [23, 26]. Instead of being considered as a self-contained platform that happens to be installed on a desk and its reach limited to the extent of its screen, it should be considered as a tool part of the whole toolset laid onto the desk, aware and capable of interacting with its surroundings.

We propose to leverage the potential of tangible interaction, while relying on the efficiency of standard desktop environments, in an integrated way. We present Tangible Viewports, a screen-based tool enabling the use of tangible objects in a standard desktop-based workflow. Contrary to many tangible user interfaces where tokens and generic props are used, we use the physical objects as canvases that are also the results of the creative or visualization process. These objects are augmented with Spatial Augmented Reality (SAR) [24], a technique consisting of displaying content onto the real world by the mean of projectors. SAR is especially well suited for creating a hybrid work environment where digital workspaces can be combined with physical tools and canvases [23, 26]. It is possible to interact with these objects through direct touch or using tools. When held in front of the computer screen, the mouse cursor can seamlessly slide from the screen *onto* the surface of the objects and interaction with native desktop application becomes possible. For example, one can use a painting software to paint over the surface of the object as if it was part

of the screen using the mouse cursor. From the viewpoint of the user, the object behaves just as a 3D model would when rendered in a viewport on the screen with the major exception that he can i) observe the object from a different viewpoint by moving the head and ii) reach out to grab the object with his hands and manipulate it freely as illustrated in Figure 1.

In this work, we emphasize the use of the desktop computer screen and its relation to augmented physical objects (Figure 1). This relation has been little explored as a complementary approach to tangible tools (e.g. [18]). We suggest that it can be leveraged to create true hybrid applications that reduce the gap between highly flexible and expressive software, currently trapped inside a flat rectangular screen, and the intuitiveness and graspable nature of our environment.

The main contributions of this work are 1) Tangible Viewports, an on-screen window in which the mouse cursor interacts with real objects as if they were virtual 3D models, 2) a proof-of-concept prototype of an integrated workspace that combines augmented physical objects and native applications, and 3) the exploration of the interaction space of this hybrid work environment.

RELATED WORK

Our work is related to several overlapping research areas, which explore the interaction around augmented objects.

Spatial Augmented Reality (SAR) consists in projecting digitally controlled light onto physical objects. SAR was first introduced by Raskar *et al.* [25, 24]. It was then followed by Underkoffler *et al.* [31] where they used a projector to display spatially coherent information with the projection space and objects laid onto it. A first interactive application of SAR on 3D objects [3] consisted in “painting” an object by using a tracked stylus. Extending this work, Marner *et al.* [18] created Physical-Virtual tools for editing the appearance of a tangible object. Other works [4, 32, 5] instead focused in creating the illusion of virtual objects being part of the scene using projective texturing and anamorphosis, which consists in projecting viewpoint corrected images taking into account the geometry of the real-world scene in such a way as to create illusions for this specific viewpoint.

Tangible User Interfaces (TUI) [12, 11] are often found along SAR. TUIs are concerned with providing tangible (i.e. physical) representations to digital information and controls. Systems using both TUI and SAR include tools for editing the appearance of physical objects [3, 18], sculpting [22, 17] and educational purposes [7]. TUIs are especially well suited for collaborative tasks and provide a strong situatedness [28]. The previously mentioned approaches are deeply rooted in physicality and stay away from traditional computer environments. Closer to our work, Akoaka *et al.* [1] created a platform for designing interactive augmented objects using either natural interaction or a standard desktop computer.

Augmented and smart spaces are systems that use see-through augmented reality or projectors, often in an office environment, to enhance the workspace. Raskar *et al.* [23] proposed the idea of a hybrid workspace that would combine the physical environment with a spatially augmented display system in

order to create a continuous mixed-reality space. Similarly, Augmented Surfaces [26] is a system that creates interactive surfaces on a table, wall and laptop using projectors. Users could use their mouse cursor to drag information between the different surfaces. Moving towards desktop systems, Kane *et al.* [15] present a hybrid laptop-tabletop system that uses two pico-projectors mounted to a laptop computer to add interactive areas on the table around the device. The system is able to detect tangible objects on the table but does not augment them in any way. HoloDesk [9] is a situated see-through display where virtual and tangible objects can be manipulated directly with the users’ hands, but does not integrate any traditional computer-related tasks. Closest to our work is the Skin system created by Saakes [27]. It consists of a workspace aimed at designers interested in materials for tangible objects. It uses a naive projection setup – i.e. no tracking and no automatic “mapping” of the textures on the objects – where textures, previously captured using a camera, are reprojected on tangible objects. We are instead interested in reducing the gap between desktop-based tools and the use of tangible objects, which also make the exploration of dynamic mediums possible – e.g. animations and programmed behaviors.

Pointing in SAR [8] is related to pointing in other contexts, namely *multi-display environments (MDE)* and *stereoscopic displays*. It is possible to compare SAR to MDE in that the tangible objects that are being augmented acts like multiple continuous (and potentially small) displays. Works most related to Tangible Viewports include Ubiquitous Cursor [33], which uses the geometry of the room to display the cursor when transitioning from two different screens, and Perspective Cursor [20, 19], which takes into account the user’s position in relation to the screen in order to give the illusion of a coherent movement from the user’s viewpoint. We directly use the principles of Perspective Cursor in our work, the main difference being that we use the cursor on more complex 3D shapes instead of being limited to planar displays. Pointing on a stereoscopic display has been studied by Teather and Stuerzlinger [30] where they evaluated different cursor types in what is effectively a “2.5D”, or projected pointing task. Again, we use a similar pointing technique but use real objects instead of virtual ones. Beyond the benefit of tangibility, using SAR also does not require the users to wear any glasses or cause discomfort due to the vergence-accommodation conflict as it is the case when using stereoscopic technology.

Our work contributes to the state of the art by leveraging the flexibility and power of desktop-based tools while still benefiting from tangible interaction, in a seamless manner.

CREATING A SEAMLESS HYBRID SPACE

The general idea of our system is to embed physical objects within the standard desktop paradigm. In our approach, we consider the screen and chosen physical objects on the desk as spatial canvases where digital information can be displayed. This design differs from other approaches (e.g. [26]) that extend the reach of the cursor to the environment, while we bring the physical objects within reach of the screen cursor. The transition from a 2D cursor located inside the screen to a cursor following the 3D geometry of the surrounding physi-

cal environment requires a change in paradigm for the users. Instead, we embed the object *in front* of the screen to reduce this change of context, as illustrated in Figure 2. This design choice is supported by studies that have shown the very low performance drop for focal depth changes compared to angular movements [29, 6]. In our current system configuration, normal use causes shallow depth of scene ($< 50\text{cm}$ – a working space created by a typically recommended distance between the user and the screen) and users are not required to rotate the head position.

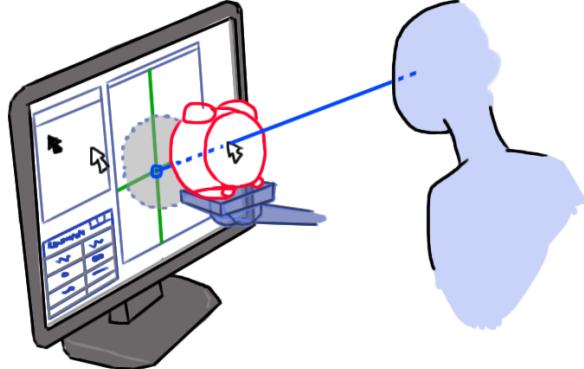


Figure 2. The user can interact with the augmented object when located in front of the screen, emulating the behavior of a 3D viewport.

Spatial Augmented Reality Setup

Our SAR setup is comprised of an augmented desktop environment and physical objects that can be brought in front of the screen. The objects can be manipulated freely by the user, or they can be placed on a support for convenience. Figure 3 illustrates the setup. The projector handling the augmentation is located behind the user, and oriented so that its vertical field of view would span from the edge of the desk up to the top of the screen. It only emits light towards the physical object, so it does not perturb the visualization of the screen. The augmentation is generated by creating a virtual counterpart of the physical scene (augmented objects and screen). Then, we create a virtual camera reproducing the projector's view of the 3D scene, using the projector's position (extrinsics) and projection cone (intrinsics). The view of this virtual camera is then reprojected onto the physical environment.

We use 3D printed objects created using a MakerBot Replicator 2 in white PLA plastic with a precision of $\pm 0.2\text{mm}$. Alternatively, we could use already existing or sculpted objects given that they would require 3D scanning before, using KinectFusion [21] for example.

The main program handling the whole installation is written with the creative coding toolkit vvvv and rendered using DirectX. The video projector is a LG PF80G with a resolution of 1920×1080 pixels calibrated using OpenCV's camera calibration functions. The tracking of the objects is achieved using an OptiTrack Trio camera, which tracks small reflective markers. Our initial version of Tangible Viewports had these markers attached to a magnetic base that could be connected to a Manfrotto articulated arm. This allows user to comfortably

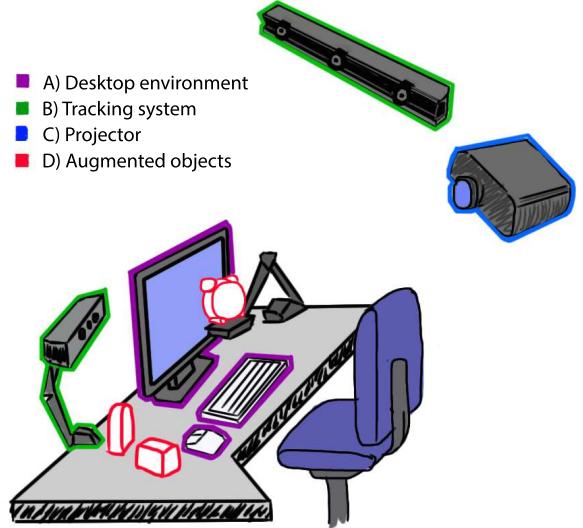


Figure 3. The SAR installation: A) The desktop environment using a standard screen and input devices, B) 6DoF tracking system (OptiTrack Trio and Microsoft Kinect v2), C) Projector and D) Physical objects that are being augmented.

position the object in 6 Degrees of Freedom (DoF) in front of the screen. We later put the markers on the objects themselves so that they could be handled more easily. Magnets were then glued directly under the objects, enabling them to still be connected to the articulated arm for longer working sessions. The tracking runs at 120 FPS with an overall latency of 8.3ms and a precision of 0.8mm. It is important that the OptiTrack system shares an overlapping field-of-view with both the Kinect and the projector. The whole implementation runs at 30 FPS (not optimized) using a 3.6 GHz Core i7 PC with Windows 8 equipped with two GeForce GTX690 graphic boards.

Cursor Handling in Tangible Viewports

The key element on which our system relies is to create the illusion that a physical object is entirely part of the screen space when located in front of it. In order to do so, we ensure that the cursor movements inside the working area occur in a continuous way, independently of where this cursor is displayed (screen or tangible viewport). The user thus perceives the visual space as a whole.

A window dedicated to the interaction with the object is created on the screen and its position is retrieved by using the Windows API. The screen is also tracked in world space by the OptiTrack system. Thus, knowing the 2D cursor position in the viewport space allows us to infer its position in world coordinates. A virtual camera is created to reproduce the user's view of the window (and whichever augmented object located in front of it). The user's head position is obtained by Kinect v2 skeleton tracking. As soon as a physical object starts occluding the screen's cursor for the observer, a 3D cursor appears at the correct location on the object, as illustrated in Figure 4. This is done by raycasting in world-space over the virtual scene from the user's viewpoint to the screen's cursor position (Figure 2). We thus obtain the 3D position

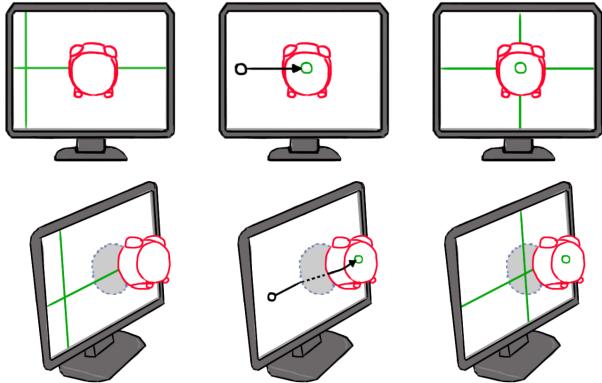


Figure 4. (Top Row) From the point of view of the user, the cursor behaves as if the object was part of a 3D viewport. (Bottom row) Side view showing the actual behavior of the mouse cursor, “jumping” from the screen onto the object when being occluded by the object from the user’s viewpoint.

and orientation on the first element on the line of sight of the user. The resulting transformation is then applied to the 3D cursor, which is displayed as a small disk aligned with the local surface’s normal. This cursor is rendered as part of the virtual scene and reprojected onto the augmented object. On the screen, a horizontal and a vertical line passing through the cursor position are displayed for enforcing the link between the tangible viewport and the screen.

In the end, this technique is fully transparent to the users. Users work with Tangible Viewports as they would do with any standard application. It is also to be noted that the head position of the user only impacts the *behavior* of the cursor; the cursor’s appearance and the augmentations on the object are completely viewpoint independent. This is especially important for collaborative settings.

Direct Touch and Gestures

Beyond cursor interaction in front of the screen, direct touch on the objects is also supported. This is achieved by attaching a small reflective marker to a ring on the user’s finger or on a tool (e.g. pen) so that it is detected by the OptiTrack system. We also tested the use of the Leap Motion in order to avoid instrumenting the finger of the user. However, the Leap is unable to detect direct touch and is better suited for fine gestures *near* the object. For coarse gestures, the hand tracking of the Kinect API is sufficient.

INTERACTION

In our hybrid workspace, interaction can either take place on the screen, on the augmented object, or on both display supports at the same time. In the following sections, we explore the interaction space by describing examples of techniques that we developed for each of these categories (Table 1).

Screen

Because our objective was to conceive a system that benefits from the advantages of standard desktops, all the usual techniques designed for such environments can directly be used.

	MODALITY		
	Mouse and Keyboard	Hybrid	Touch and Gesture
LOCATION	Widgets, native applications, programming	—	Touch screen based interfaces*
Hybrid	Drag and drop, hybrid widgets	Pick and drop, object annotations, data visualization	Gestural control of virtual version
Object	Pointing on objects	Bimanual interaction	Navigation, tangible design

* Out of the scope of the paper

Table 1. Interaction space around Tangible Viewports.

Widgets

We have designed a custom application based on such standard widgets for modifying the appearance of an augmented object (See Figure 5). For example, selecting the background color of an augmented object can be done directly by way of a color palette. This application served as a basis for the evaluation of the system that we present later in this paper.

Native Applications

It is also possible to use native professional applications. As an example, we linked the output of Adobe Photoshop, a software that is ubiquitously used in the design and artistic industries, to our system. Hence, we leverage the skills that professionals already acquired with these tools. The most straightforward use is UV painting (Figure 5(d)) which consists of adding graphics on a 3D model. It is a task that can be done either in a 2D painting environments using a UV layout or directly on a 3D view of the object. Both can be achieved using Photoshop. We retrieve the texture that is being painted in real-time and update the augmented object accordingly. Every time an operation is performed on the design, the physical object’s appearance also gets updated. This can be especially useful in object design, where the final result is not a 3D render but an actual object.

Programming

In addition to the connection of existing tools, we also included Creative Coding capabilities. In practice, creative coding is often comprised of programming toolkits that are focused on visual results and short feedback loops. For these reasons, it is often used for prototyping. Examples of such toolkits include Processing, OpenFrameworks and vvvv. These approaches make it possible to envision a near future where physical objects are comprised of thin and flexible screens [10] and with which users could tinker with their augmented content (dynamic appearances and behaviors). Programming is an activity that is almost exclusively conveyed on standard computers. It is possible, then, to create a program and visualize its execution in real-time on a tangible object. As an example, we created a simple program where the appearance of a clock evolves with the time. The results of this program can be visualized directly on an augmented physical clock (Figure 5(c)).

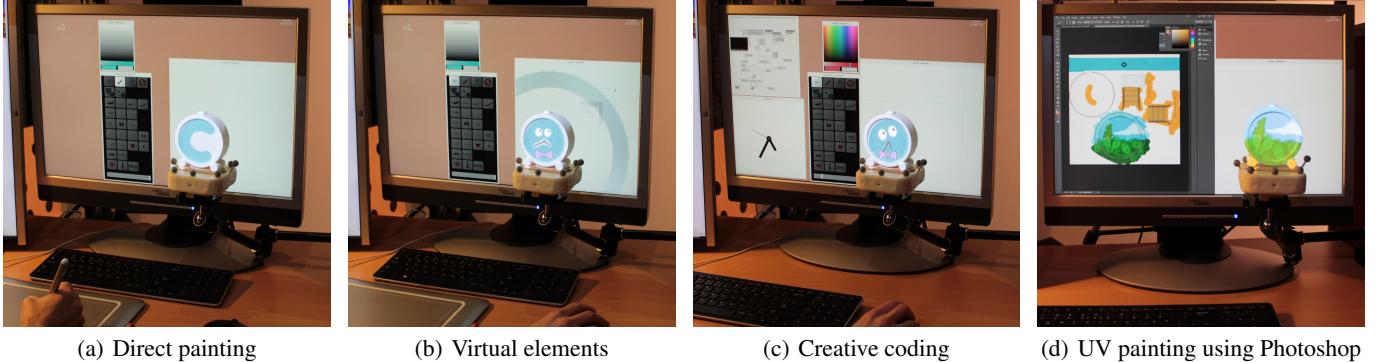


Figure 5. Different features to modify the appearance and behavior of the physical object.

Physical Object

This section presents the interaction techniques we have implemented to support the use of physical objects: direct interaction, pointing on object using the tangible viewport window and bimanual interaction.

Direct Interaction

Working with physical objects has the benefit of enabling manipulation directly with the hands. No 2D to 3D mapping operations are required to create a desired point of view as is required in desktop 3D applications. Also, since the augmentation occurs on the surface of the object, changing the viewpoint can simply be achieved by moving the head. The user can thus observe the object in a natural way, which radically differs from what he or she is used to do with a virtual version of models displayed on flat screens. Also, direct touch can be used whenever precision or specific tools are not required. For example, when creating interactive objects, one can use interactors or trigger behaviors directly, similar to [1].

Pointing on Objects

In addition to direct manipulation of the tangible objects, our system supports cursor-based indirect interaction for completing interaction tasks directly onto the physical objects. These tasks can be pointing, drawing, selecting or moving virtual elements. Compared to an approach where the user would interact directly on the physical object, indirect interaction offers several complementary advantages. It does not require specific input devices, it is fast and accurate, less prone to fatigue, and it integrates within the desktop workflow.

Bimanual Interaction

Handling the physical object and using the mouse can be achieved at the same time following a bimanual interaction approach [2]. The hand holding the object plays the role of reference frame and assists the dominant hand which is dedicated to fine mouse movements. This approach leverages the precision and stability of 2D pointing and the easiness of 6 degree of freedom manipulations of 3D objects.

Hybrid Screen/Object

Both the physical objects and the screen are part of the same working space. Consequently, it is possible to directly link operations on the screen with actions on the physical objects.

The converse is also true. We present application examples that use both object and screen simultaneously.

Drag and Drop

Since the viewport creates a seamless continuum between the screen and the object, drag and drop operations can be used with the mouse cursor. This operation would not be possible using touch and would have to be replaced by pick and drop.

Hybrid Widgets

The standard approach for applying transformations (e.g. scaling and rotation) to visual elements displayed on a screen is to use widgets centered on these elements. The problem with standard SAR setups is that, although technically possible [5], it is very difficult to create the illusion of floating visual elements around the object as soon as no material can support the display. We designed hybrid widgets that are operated on screen. We reproject the position of the selected element on the screen based on the user's viewpoint and we place 2D widgets centered on this location. When moving the physical object, the position of the widgets is updated accordingly on the screen. These transformation widgets that allow the rotation and scaling of the selected element are illustrated in Figure 5(b). They are relatively big and they do not touch directly the physical objects. This design choice has been made to avoid problems of eye accommodation between the depth of the object and the depth of the screen. Hence, after selecting an object to modify, users can quickly grab and manipulate the widgets, without eye fatigue.

Object Annotation

Another opportunity offered by the fact that a screen stands behind the physical object concern the display and the entry of text. Indeed, these operations may be difficult to complete in many traditional SAR setups. In our case, it is easy to annotate a physical object by selecting an anchor point (either with the mouse or direct touch) and typing a related note being displayed on the screen, with the keyboard. Inversely, one can select a note on the screen, and see the corresponding area directly on the physical object. This creates a text box positioned in an empty zone of the screen which is linked to the projected position on the screen of the anchor point.

Physical Data Visualization

Beyond annotations that can benefit to many fields (e.g. inspection of manufactured objects), we have explored the use of a hybrid approach in the scope of data visualization. Data visualization (and especially 3D data visualization) has been shown to gain from a physical representation [13]. Using the tangible viewport window, it is possible to add interactivity to physical visualization. In particular, to query more information on some aspects of the visualization, one can just point at the area of interest to see related data on the screen, or she or he can select an entry on the screen to see the corresponding elements on the physical visualization (see Figure 6(a)).

Synchronized Views

We also explored the synchronization between a virtual version of an object displayed on screen and a physical one. When the tangible object is not in front of the screen, the tangible viewport window displays a virtual version of the augmented object (Figure 6(c)). Modifying the virtual version updates the tangible version in real-time.

Being able to have two representations of an augmented object, one on screen and one physical opens possibilities, namely for collaboration. For example, it would be possible to expose the view of a user handling the physical object or providing advanced visualizations such as a heat map of touched areas (Figure 6(d)). Also, multiple users can have their own duplicated augmented object (Figure 6(b)). These users can be working either locally or remotely.

The synchronization between real and virtual can be paused, for example using a gesture (e.g. pulling the object rapidly away from the window), to compare multiple versions. Bringing back the physical object in front of the window merges the two versions on the physical canvas.

ILLUSTRATIVE SCENARIO

To illustrate the use of Tangible Viewports, we describe here an object design scenario where an artist is experimenting with new visual design ideas for a product (see complementary video).

She can start by sketching first ideas on a paper, and then use a modeling tool to create a 3D sketch. Equipped with a 3D printer, she can print one (or many) physical objects to have in front of her. She can first directly paint on the object using the mouse cursor. Then, she can use a digital painting application such as Adobe Photoshop or a vector graphics editor like Inkscape to draft a logo on her computer. Then, using the mouse, seamlessly drag the logo from the editor directly to the physical prototype she just printed. The prototype can be physically manipulated to review the appearance. Modification to the design on the desktop computer will be automatically reflected in real-time onto the object. She can scale and rotate the logo directly on the physical object to see directly the impact of her modifications. This way, the feedback loop between the design activities (which require specialized software) and the validation of the effect it has in physical form can be greatly reduced. If required, new versions of physical objects can iteratively be 3D-printed, as we currently do with 2D printers when working on 2D documents.

By making the interaction with the physical objects coherent with the traditional way of manipulating 3D information on a desktop computer, it is possible to leverage the experience of users with their professional tools, while at the same time adding the richness of tangibility and physical visualization.

In this scenario, we can also imagine one or several collaborators participating to the design choices. These collaborators can directly observe and manipulate the augmented object, and ask the main designer to update the design in real time. This kind of social collaboration is harder to obtain with traditional design tools.

USER FEEDBACK AND DISCUSSION

We conducted an exploratory study where we asked participants to manipulate a preliminary version of the system, as well as a non-tangible version of the tool. The objective of this study was to assess how physical objects integrate within a standard screen space. We have designed a simple custom creation tool (see Figure 5(a) and 5(b)) for this purpose. Fourteen participants (9 males, 5 females, mean age 25.6 ($SD = 3.7$)) took part in this study. Half of the participants started the experiment with Tangible Viewport, then they moved to the non-tangible one, and half did the opposite. In both cases, participants were introduced to the main features of the tool, and the experimenter explained what was expected from them. Participants were asked to create a personal visual design of a clock. The only difference between the two versions of the tool is that, in the tangible version, the results of the creation was directly displayed on a 3D printed clock, whereas the virtual representation of the model was used in the standard viewport version. For changing the view on the object, subjects had to either manipulate the object and/or move the head “naturally” with Tangible Viewport, whereas they were using a trackball metaphor operated with the mouse middle-button in the standard viewport version, as commonly done in standard desktop 3D tools.

Subjects were asked to follow a tutorial for customizing their clock (Figure 7), which included: 1) choosing a background color and painting the front face, 2) adding virtual elements and resize/rotate them, and 3) making a drawing on the side and back of the object. This scenario was designed to ensure that the main features of the tool were used under different conditions. For example, Step 3 tests the ability of the participants to draw freely on curved surfaces.

After the experiment, participants were asked to answer two questionnaires: the User Experience Questionnaire [16] and a custom questionnaire aiming at obtaining user feedback about the usability of the tested systems (5 points Likert scale) and their preferences between the two. Both questionnaires showed no significant difference between the two versions of the system. They were also invited to leave comments and feedback about what they liked and disliked about each version of the system. Overall, the majority of the participants preferred manipulating the tangible version (12 out of 14) and were more satisfied with the final result (11/14). No participant mentioned difficulties moving from the screen to the object. These results seem to indicate that the tangible viewport metaphor works well, and it is comfortable to use.

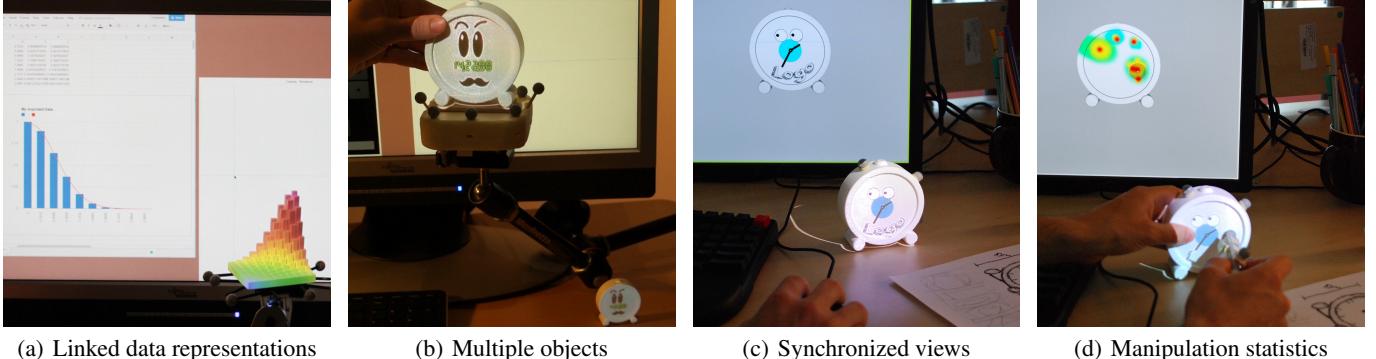


Figure 6. Different synchronization modes between virtual on-screen and tangible versions.

Regarding the comments, among the most appreciated features spontaneously cited by the subjects is the ability to work with a real object (9/14) and to have a physical view on the final product (6/14). For example S1 liked that “*you can see the real object with the elements you draw. That way, you can observe the final product before it is produced*”. S9 mentioned that “*The creation feels much less virtual*” and that “*going from the screen to the object is fun*”. A few participants also insisted that they liked to be able to manipulate the object with their hands (5/14), while others found the magnetic base uncomfortable (5/14) – which is why we later replaced the base and attached the markers directly onto the objects. Complaints were made (5/14) regarding the fact that the editor lacked important features such as “undo” and “magic wand”. This highlights that the interaction between the screen and the object was working well enough that the main focus was about the painting features.



Figure 7. Examples of participants' creations using Tangible Viewports.

Regarding the technical solution, several participants (6/14) mentioned that the augmentation calibration was not precise enough, which could be improved by using more advanced known solutions such as the one used by Jones *et al.* [14]. They also explicitly mentioned some delays and robustness issues on the head tracking (5/14). The second iteration of the system corrected these issues by replacing face tracking by skeleton based head tracking and better Kinect positioning. Regarding the cursor, some participants (4/14) did not like the fact that changing the head position was moving the cursor on the object, a side effect of using perspective cursor. This issue could be addressed with a system that would prevent the cursor on the object to move when the head position of the user changes and instead correct the on-screen cursor's position when it reaches the edge of the object's silhouette, from the user's point of view. Such alternatives will be studied on the future.

CONCLUSION AND FUTURE WORK

In this paper, we have introduced Tangible Viewports and we have described an effective implementation of this concept. A preliminary study showed that the overall usability of this system is good. Tangible Viewports do not aim at replacing existing systems. Indeed, we have shown that, from a technical and user point of view, the seamless integration of physical and virtual tools is not just feasible but enriches both.

One of the current limitations of a tangible approach is the rigidity of the physical elements, which cannot (yet) be reshaped in real time. Our vision is that 3D printing will become as efficient as 2D printing in a near future. Hence, one will be able to use the flexibility of virtual elements to explore variations of geometries, and use physical elements as soon as he or she will require a perception of the created shape that goes beyond a simple rendering on a 2D screen.

In the future, we would like to assess more precisely how Tangible Viewports may leverage creativity in professional uses. This will require dedicated user studies with targeted users for investigating in more depth on how interactive physicality impacts performance. Finally, Tangible Viewports leverage the desktop computer capabilities as part of a set of tools to interact with physical matter. In the future, we plan go further, by merging the desktop on the workbench itself.

ACKNOWLEDGMENTS

This work was supported by the ISAR project ANR-14-CE24-0013. We would also like to thank the reviewers for their constructive feedback and all the participants who tested the system.

REFERENCES

1. Eric Akaoka, Tim Ginn, and Roel Vertegaal. 2010. DisplayObjects: Prototyping Functional Physical Interfaces on 3D Styrofoam, Paper or Cardboard Models. In *Proceedings of the Fourth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '10)*. ACM, 49–56.
2. Ravin Balakrishnan and Gordon Kurtenbach. 1999. Exploring Bimanual Camera Control and Object Manipulation in 3D Graphics Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '99)*. ACM, 56–62.

3. D. Bandyopadhyay, R. Raskar, and H. Fuchs. 2001. Dynamic shader lamps : painting on movable objects. In *Augmented Reality, 2001. Proceedings. IEEE and ACM International Symposium on*. IEEE Comput. Soc, 207–216.
4. Hrvoje Benko, Ricardo Jota, and Andrew Wilson. 2012. MirageTable: Freehand Interaction on a Projected Augmented Reality Tabletop. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, 199–208.
5. Hrvoje Benko, Andrew D. Wilson, and Federico Zannier. 2014. Dyadic Projected Spatial Augmented Reality. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, 645–655.
6. Jessica R. Cauchard, Markus Löchtefeld, Pourang Irani, Johannes Schoening, Antonio Krüger, Mike Fraser, and Sriram Subramanian. 2011. Visual Separation in Mobile Multi-display Environments. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology (UIST '11)*. ACM, 451–460.
7. Jérémie Frey, Renaud Gervais, Stéphanie Fleck, Fabien Lotte, and Martin Hatchet. 2014. Teegi: Tangible EEG Interface. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, 301–308.
8. Renaud Gervais, Jérémie Frey, and Martin Hatchet. 2015. Pointing in Spatial Augmented Reality from 2D Pointing Devices. In *INTERACT*. Bamberg, Germany, 8.
9. Otmar Hilliges, David Kim, Shahram Izadi, Malte Weiss, and Andrew Wilson. 2012. HoloDesk: Direct 3D Interactions with a Situated See-through Display. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, 2421–2430.
10. David Holman, Audrey Girouard, Hrvoje Benko, and Roel Vertegaal. 2013. The Design of Organic User Interfaces: Shape, Sketching and Hypercontext. *Interacting with Computers* 25, 2 (2013), 133–142.
11. Hiroshi Ishii, Dávid Lakatos, Leonardo Bonanni, and Jean-Baptiste Labrune. 2012. Radical Atoms: Beyond Tangible Bits, Toward Transformable Materials. *interactions* 19, 1 (jan 2012), 38–51.
12. Hiroshi Ishii and Brygg Ullmer. 1997. Tangible Bits: Towards Seamless Interfaces Between People, Bits and Atoms. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems (CHI '97)*. ACM, 234–241.
13. Yvonne Jansen, Pierre Dragicevic, and Jean-Daniel Fekete. 2013. Evaluating the Efficiency of Physical Visualizations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, 2593–2602.
14. Brett Jones, Rajinder Sodhi, Michael Murdock, Ravish Mehra, Hrvoje Benko, Andrew Wilson, Eyal Ofek, Blair MacIntyre, Nikunj Raghuvanshi, and Lior Shapira. 2014. RoomAlive: Magical Experiences Enabled by Scalable, Adaptive Projector-camera Units. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology (UIST '14)*. ACM, 637–644.
15. Shaun K. Kane, Daniel Avrahami, Jacob O. Wobbrock, Beverly Harrison, Adam D. Rea, Matthai Philipose, and Anthony LaMarca. 2009. Bonfire: A Nomadic System for Hybrid Laptop-tabletop Interaction. In *Proceedings of the 22Nd Annual ACM Symposium on User Interface Software and Technology (UIST '09)*. ACM, 129–138.
16. Bettina Laugwitz, Theo Held, and Martin Schrepp. 2008. *Construction and evaluation of a user experience questionnaire*. Springer.
17. Michael R Marner and Bruce H Thomas. 2010. Augmented foam sculpting for capturing 3D models. In *3D User Interfaces (3DUI), 2010 IEEE Symposium on*. IEEE, 63–70.
18. Michael R. Marner, Bruce H. Thomas, and Christian Sandor. 2009. Physical-virtual tools for spatial augmented reality user interfaces, In *Mixed and Augmented Reality, 2009. ISMAR 2009. 8th IEEE International Symposium on. 2009 8th IEEE International Symposium on Mixed and Augmented Reality (oct 2009)*, 205–206.
19. Miguel A. Nacenta, Satoshi Sakurai, Tokuo Yamaguchi, Yohei Miki, Yuichi Itoh, Yoshifumi Kitamura, Sriram Subramanian, and Carl Gutwin. 2007. E-conic: A Perspective-aware Interface for Multi-display Environments. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology (UIST '07)*. ACM, 279–288.
20. Miguel A. Nacenta, Samer Sallam, Bernard Champoux, Sriram Subramanian, and Carl Gutwin. 2006. Perspective Cursor: Perspective-based Interaction for Multi-display Environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*. ACM, 289–298.
21. Richard A Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J Davison, Pushmeet Kohi, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. 2011. KinectFusion: Real-time dense surface mapping and tracking. In *Mixed and Augmented Reality (ISMAR), 2011 10th IEEE International Symposium on*. IEEE, 127–136.
22. Ben Piper, Carlo Ratti, and Hiroshi Ishii. 2002. Illuminating Clay: A 3-D Tangible Interface for Landscape Analysis. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '02)*. ACM, 355–362.
23. Ramesh Raskar, Greg Welch, Matt Cutts, Adam Lake, Lev Stesin, and Henry Fuchs. 1998b. The Office of the Future: A Unified Approach to Image-based Modeling

- and Spatially Immersive Displays. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '98)*. ACM, 179–188.
- 24. Ramesh Raskar, Greg Welch, and Henry Fuchs. 1998a. Spatially augmented reality. In *First IEEE Workshop on Augmented Reality (IWAR'98)*. Citeseer, 11–20.
 - 25. Ramesh Raskar, Greg Welch, Kok-Lim Low, and Deepak Bandyopadhyay. 2001. Shader Lamps: Animating Real Objects With Image-Based Illumination. In *Eurographics*, StevenJ. Gortler and Karol Myszkowski (Eds.). Springer Vienna, 89–102.
 - 26. Jun Rekimoto and Masanori Saitoh. 1999. Augmented Surfaces: A Spatially Continuous Work Space for Hybrid Computing Environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '99)*. ACM, 378–385.
 - 27. Daniel Pieter Saakes. 2010. *Shape does matter: designing materials in products*. TU Delft, Delft University of Technology.
 - 28. Orit Shaer and Eva Hornecker. 2009. Tangible User Interfaces: Past, Present, and Future Directions. *Foundations and Trends in Human-Computer Interaction* 3, 1-2 (2009), 1–137.
 - 29. Desney S Tan and Mary Czerwinski. 2003. Effects of visual separation and physical discontinuities when distributing information across multiple displays. In *INTERACT*, Vol. 3. 252–255.
 - 30. Robert J. Teather and Wolfgang Stuerzlinger. 2013. Pointing at 3D Target Projections with One-eyed and Stereo Cursors. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, 159–168.
 - 31. John Underkoffler and Hiroshi Ishii. 1999. Urp: A Luminous-tangible Workbench for Urban Planning and Design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '99)*. ACM, 386–393.
 - 32. Andrew Wilson, Hrvoje Benko, Shahram Izadi, and Otmar Hilliges. 2012. Steerable Augmented Reality with the Beamatron. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology (UIST '12)*. ACM, 413–422.
 - 33. Robert Xiao, Miguel A. Nacenta, Regan L. Mandryk, Andy Cockburn, and Carl Gutwin. 2011. Ubiquitous Cursor: A Comparison of Direct and Indirect Pointing Feedback in Multi-display Environments. In *Proceedings of Graphics Interface 2011 (GI '11)*. Canadian Human-Computer Communications Society, 135–142.