Blended UI Controls For Situated Analytics

Neven A. M. ElSayed*, Ross T. Smith*, Kim Marriott† and Bruce H. Thomas*
*Wearable Computer Lab, University of South Australia
†Monash Adaptive Visualisation Lab, Monash University

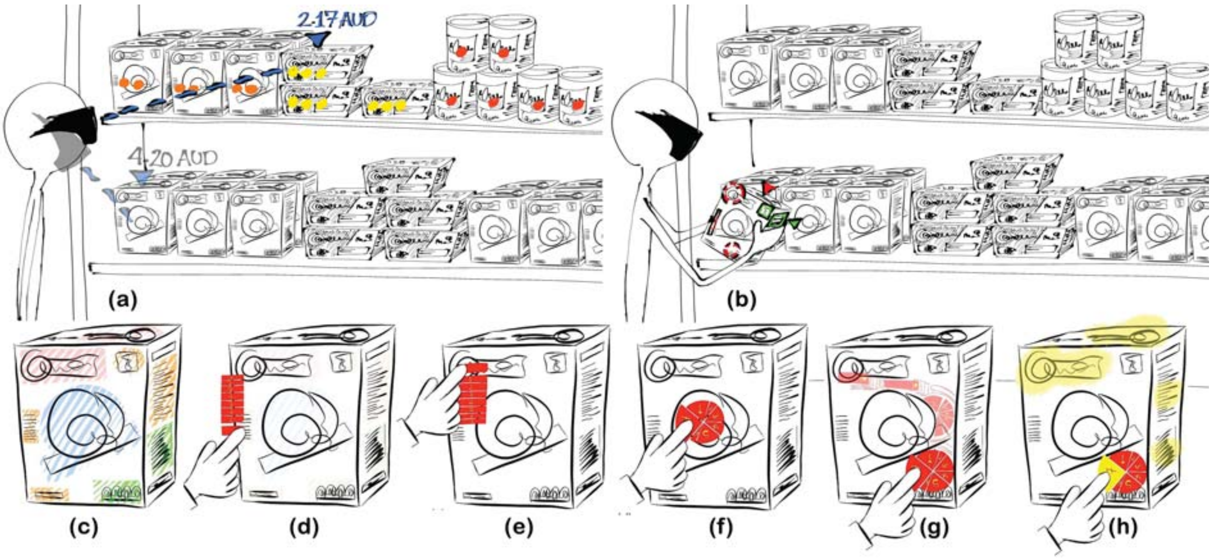


Fig. 1: Situated Analytics blended controls demonstrated in a shopping context. (a) The user is viewing products on a supermarket shelf through an augmented reality display with registered virtual annotations on physical objects. Details such as price are shown on the product the user is focusing on. (b) User grasps and interacts with the physical objects, invoking them as tangible input controls that provide contextual affordance. (c) Each physical object is associated with metadata inherited from its physical context, providing a contextual and situationally aware user interface. (d) User tapping on the text of a physical object to invoke an off-menu maintaining the contextual information. (e) Clicking on the top label on the physical object to invoke a drop-down menu, as the priority of the top label's menu was set to a higher value than the contextual information. (f) The appearance of the UI controls alter based on its assigned occupation area. (g) The UI controls can be assigned to override or avoid the physical context (h) Finally when the user selected a menu's item; the system highlighted the contextual information related to the selected item, which is calculated from the stored the metadata.

***Abstract*—This paper presents a context aware model for situated analytics, supporting a blended user interface. Our approach is a state-based model, allowing seamless transition between the physical space and information space during use. We designed the model to allow common user interface controls to work in tandem with the printed information on a physical object by adapting the operation and presentation based on a semantic matrix. We demonstrate the use of the model with a set of blended controls including; pinch zoom, menus, and details-on-demand. We analyze each control to highlight how the physical and virtual information spaces work in tandem to provide a rich interaction environment in augmented reality.**

***Keywords*—** Augmented Reality, Situated Analytics, Immersive Analytics, Interaction Techniques, In-situ Interaction, Context-driven Interaction.

I. Introduction

Situated Analytics (SA) is a new research direction that aims to provide analytical reasoning embedded in the user's physical environment. It brings together visual analytics (VA) with augmented reality (AR). One of the main challenges facing situated analytics is the difficulty of interaction [8]. This is driven by two main points: 1) The user needs to interact with physical objects (physical space) and the data associated with each physical objects (information space). 2) The SA user interfaces should work in tandem with the physical context, and alter based on the real world situation. This paper presents Blended UI Controls as a novel solution for this SA interface challenge. User interfaces to VA systems are complex and are currently focused on a traditional desktop computing environments. Blended UI controls extend this scope and are designed for mobile applications to support sense-making in the field. Applications of both these tools are employed for analysis of a diverse set of big data sources. Blended UI controls leverage the natural ability to take advantage of the user's physical context to support analytical operations.

The Blended UI Control model employs user interface controls that leverage the physical and virtual spaces for their functionality (Figure 1). The model allows Situated Analytics (SA) designers to develop controls that have a synergy between the semantics of the virtual and physical information. The appearance of the controls is dynamic depending on their placement and function on the physical object. The novelty of the techniques is their context-aware dynamic blending of physical/virtual user interface controls allowing seamless transition between the physical and information spaces.

The development of these techniques was inspired by observing a consumer behavior specialist, who is a non-AR expert, using an AR system for the first time. When presented

978-1-5090-5272-1/16/$31.00 ©2016 IEEE

with AR information registered on a biscuit package, she attempted to select physically labeled regions of the package, tapping on the physical box to obtain more information. Later, she explained that most consumers nowadays are so used to being able to interact with objects virtually (i.e. websites, phone applications, and touch screens), that they would naturally expect this function in an AR technology. As an example, when a customer holds a food item, virtual nutrition information is presented and calculated based on the ingredients of the item. Moreover, to use package's printed labels to incorporate visualizations, or navigate to external information. The consumer specialist's actions during the AR trial and her explanations highlighted the need for AR information and physical information to work in concert, not only based on the geometric shape or the spatial location [10].

Previous investigations into AR information visualization specified AR information either in a fixed location [1] (same for each object) or automatic layout to prevent occlusion [4]. Recently, the interest of SA [9] has increased, by associating data attached to the object (e.g., printed information on a package). The existing SA interaction and visualization techniques are used to overlay the physical objects and to select them digitally through an AR display.

The main contribution of this paper is our new model for interaction techniques that is more natural, intuitive, specific, informative and responsive. Printed visual information on the object is integrated into the user interface controls, allowing a two-way fusion between the physical world and the virtual controls. This model is based on the classic Model-View-Controller model adapted to solve some of the interface challenges associated with SA. The model uses semantic rules to ensure the context aware physical and virtual information is consistent with both Interaction and UI elements, to create the presentation widgets. We present exemplars using the model on both tablet and head-worn display, for various data types and application scenarios that illustrate how the model provides in-situ interactive information visualization for SA.

II. Background

This section provides an overview of relevant background literature from which this paper draws. Relevant examples of AR interactive visualization are presented, followed by a discussion about the recent immersive interaction investigations.

A. Augmented Reality Interactive Visualization

AR is a technology for the in-situ presentation of information, allowing the addition of virtual information to a user's experience of the physical world. The most common form is the overlaying of computer graphics on a user's view of the world [2]. The Touring Machine [11] was one of the early approaches that provided an interactive AR visualization tool, by employing a head worn display attached to a wearable computer to highlight key points of interest and supported interactions on a handheld tablet. Later, White et al. presented one of the early approaches to visualizing multidimensional information in AR, and their method permits users to inspect a static database. Queries are executed by computer vision techniques to identify physical objects and employed tangible interactions to inspect the data and equate solutions.

Working with abstract information increases the complexity challenge of AR interaction. White et al. [33] investigated the different types of menus that can be used for information visualization. These menus, however, were disconnected from the physical, contextual properties. Piekarski et al. [24] applied the interactivity with the controls using gloves [23], and later Veas et al. [30] applied the similar interactions by assigning the menu items to individual fingers. Another, interaction approach was developed by Slay et al. [27] an information visualization in AR, using a marker-based interaction system.

Recently, investigations have been expanded to find new immersive ways to interact (e.g. gesture and tangible interactions) with the vast amount of data associated with the AR systems. Walsh et al. [32] presented tangible, touch-based, ad-hoc user interaction controls allowing users to create and map new inputs such as sliders and radio buttons. Tangible proxemic interactions are employed to change the virtual information based on the distance between the user's view and the physical object. Marquardt et al. [21] developed a toolkit for proxemic techniques, calculating the proxemic distance between entities and alter the representation based on the calculated proxemics distance. Piumsomboon et al. [25] used extended hand gesture interaction techniques to manipulate the virtual objects embedded in the real scene, including select, move, rotate, and scale functions. Their study aims to lead to consistent universally accepted AR designed gestures.

With the increasing data that needs to be presented in AR applications, researchers started to investigate exploration techniques in AR. "Through the looking" glass considered to be one of the earliest approaches that introduced focus and context view for AR, Looser et al. [19], This approach was inspired by the magic lens technique [6], supporting object selection and manipulation, and information filtering, which is applied all on the virtual content. Later, Looser et al. [20] presented a modified version to control the presented information's dense using two hand interaction controllers. Exploration approach was introduced by Kalkofen et al. [14] for multilevel level visualizations to handle object occlusion.

B. Immersive Interaction

Recently, investigations have been interested in situation immersive interactions. With the increased amount of the data, situational awareness can present the information based on the interaction situation, reducing the data cluttering.

Leithinger et al. [18] developed a contextual awareness menu to solve this cluttering challenge of the blended interaction. It is a menu technique for table-top setup, which provides a user drawing menu. The technique showed better results compared with the traditional drop-down menus. Body interaction is another approach being used for large and spatial displays. Schmidt et al. [26] have proposed an interaction technique, enabling visualization altering based on the body's pose and location using a Kinect for tracking.

Recently, motion gestures have become a potential approach for of mobile and tablets using the device motion's as input parameters, providing smooth and continuous input interaction Oakley et al. [22] used the motion interaction concept for menu selection, with 90-degree rotation around the horizontal axis, by using TiltControl motion tracking. Another technique was introduced by Baglioni et al. [3] to support eight tilting gestures with 6DOF. The two motion gestures techniques proved that motion input might reduce the cluttering factor resulted from overlapping interaction and visualizing functionalities on the same zone. The continuous interaction primary goal is to fly through the visual representation, such as scene rotation, zoom, and pan operations.

Existing solutions for AR interaction techniques provide users with a limited number of predefined interaction perspectives for the presented data and the input controls are either static for all objects or have a limited number of controls that can be associated. Working with abstract information in AR requires more methods of interaction than the traditional approach, allowing the user to manipulate the data freely and explore their relationship, in the two spaces: physical and information. A recent direction of research is Situated and Immersive Analytics [8], [10], [7], and blended spaces [5], redresses this focus by examining how best to support interactive visualization techniques on immersive visualization platforms. Subsequently, Neven ElSayed et al. [10] proposed Situated Analytics (SA) as a model of interactive information visualization in AR. They derive SA from the domains of Visual Analytics and Augmented Reality to support a new form of in-situ interactive visual analysis. However, not only is the interaction the challenging part, the input controls that use these interaction techniques are also problematic.

III. Blended User Interface Controls

Situated analytics techniques for information visualization [8], [10], [9] enable users to interact with physical objects analytically and to manipulate their associated data. This paper presents the blended user interface control's concept, which fuses the controls into the physical objects, and drives the controls' appearance from the physical context, affording dynamic widget appearance and layout techniques.

Figure 1 shows a blended user interface control example in a supermarket context. The user is viewing a supermarket shelf through an AR display, seeing the overall nutrition through the virtual annotation (Figure 1-a). The virtual annotations are associated with each physical object and adapted based on the size and shape. The blended system also provides more detailed information such as the price overlaid with the AR functionality. We did not wish to use any traditional cursor control devices, such as a handheld or body worn mouse [28].

The user selects and picks up a product from the shelf, exploring more detailed information (Figure 1-b). The picking up action automatically invokes a "details on demand" exploration mode, using the distance between the AR camera and the physical object. In the "details on demand" blended view, the contextual features on the product's box are converted to be interactive regions (Figure 1-c). Figures 1d-h illustrate a user interaction with a blended menu, on a physical object. The blended menu's appearance is calculated based on the spatial and contextual features on the physical box. Figure 1-c depicts the authorized "regions of interest" that have been stored with each physical object, illustrated with different colors in the figure. Each region is associated with "semantic matrix", holding the region's metadata and permissions. Figure 1b shows a user physically clicked on one of the regions, invoking a menu. The menu's alignment is based on the interaction location and the region matrix. Figures 1e, 1f, and 1g depict the menu appearance changing when the user drags the menu onto different regions of the physical object. Figure 1e depicts a menu deformation example by not occluding the "product name" because of the region's occlusion permissions was set to "false". When the user selects any menu item, the system highlight the "regions of interest" that is related to the selected menu item (see Figure 1h). This scenario illustrates two important concepts. Firstly, the uniqueness of the blended techniques. Secondly, the importance of the semantic matrix, and its facilitation of the association of control and presentation. This context-aware association process leads to the synergy between both the physical and virtual information that enhances the understanding of information and UI controls.

Figure 1 illustrates the benefit of using blended controls for situated analytics, highlighting how the physical and virtual information work in synergy. The previous SA approaches were implemented based on the well-known ModelView-Controller (MVC) [16] and inspired by the tangible model view controller [13]; the previous SA approaches do not support Blended UI Controls, as the input to the controller is decoupled from the view (display output). By the definition of Blended UI Controls, there must be a coupling between the physical and virtual content to support the interactions in Situated Analytics blended controls. The difference between the MVC and our adapted Blended UI Controls Model is the MVC isolates the input and output devices from eachother. In the original MVC, the input devices are connected to the controller and the output devices from the view. In the Blended UI controls model the physical objects are supported through the blending bond to accommodate both input to the controller and the output from the view. The blended bond allows for changes in the view through physical interaction that is not mediated through the controller. The traditional MVC can not show the required representation to support physical based interactions for areas such as Situated Analytics [9], SAR [31], [29] and Immersive Analytics [7] that incorporate real-time tangible interaction and interactive data representation.

IV. Blended User Interface Controls Model

This paper proposes a Model-View-Controller for blended SA, which has an awareness of the physical context. Based on Krasner and Pope [16] the controllers and views are independent, as any of them can be replaced without affecting the system logic or data manipulation. Krasner and Pope presented model was targeting the software design. Later, Ishii has presented a tangible Model-View-Controller [13], breaking down the view into two representations, the physical representation (Rp) and the digital representation (Rd). Ishii's adaptation focuses on the data representation, and how the tangible interactions manipulate the view. Ishii adaptation, however, did not support the blending for the input/output, as the tangible controls manipulate the data in the model, while our blended controls manipulate the data in the view (in particular the visual component of the UI components). Blended controls are more than influence passing between the view and the controller, the output of both, became one bonded piece, which we called the *blending bond*.

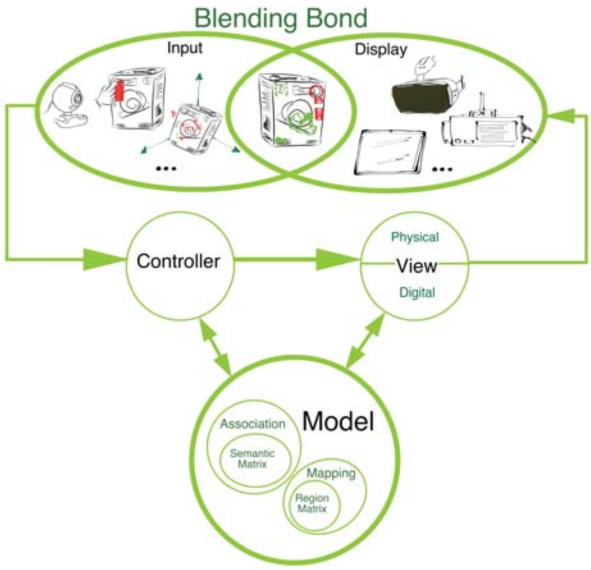


Fig. 2: Blended UI Controls Model, a blended adaptation of the original Model-View-Controller by Krasner and Pope [16]

This section explains the blended controls four main components: *Model, Controller, View, Blending Bond*. Figure 2 shows our blended adaptation of the Model-View-Controller highlighted in green color.

Model manages the data driven application from information on the physical objects, containing two main processes: *Mapping* and *Association*. The Mapping process creates the *Region Matrix*. The Region Matrix stores the data and their spatial location for physical objects. Each physical object has a number of data points associated with its contextual features. The *Association* process creates the *Semantic Matrix*, containing the properties and constraints of the UI elements that are attached to a data point, which we call it "region of interest."

Controller contains the in-situ techniques that can be used as follows: to interact with the physical object, its associated information, or to invoke a UI control. The novelty of the interaction components is that it updates both the Semantic Matrix and the Region Matrix. The blended controller is displayed at the same time. The fusion between the controller and view ensures a two-way information flow between the physical and virtual spaces to make the physical objects part of the interaction and providing state-awareness interaction transition. The blended controllers allow for the use of physical objects as tangible controllers and for the use of contextual features as physical GUIs.

View contains a set of blended user interfaces and visualization for dynamic widgets creation. The properties and position of the UI elements are controlled through the model, which is updated through the controls. This separation between the UI and the contextual awareness parts of the model simplifies the design process of UI elements. The view output is a fusion between the contextual view of the physical objects (View-Physical) and the augmented representation (View-Digital).

Blending Bond components work together to register the virtual information in the physical space, based on the meaning of the information. The user's understanding is accumulated based on the feedback loop between the controller and view components, through the interactive blended session. Moreover, the physical affordance can be a potential solution for state model systems. In the next sections, we explain the elements of the model and detail how physical space's operational interactive information visualization is enabled.

A. *The Model*

The blended model is the main contribution of our approach. This component allows a two-way, real-time association between the physical and the virtual information, enabling contextual and situation awareness for the interactive information process. As previously mentioned, this component consists of two main processes: *Mapping and Association*.

The Mapping process determines the "regions of interest" on the physical objects, which can be performed as follows: manually, through image processing techniques, or by using sensors attached to particular locations on the physical object. The mapping information is stored and updated in the *Region Matrix*. The Region Matrix holds a key for each mapped region and its spatial coordinates on the physical object. The Region Matrix can be updated in real-time using an authoring tool.

The Association process is used to assign metadata to each "region of interest" in the Region Matrix. This metadata is driven by the physical context, which is initially stored using authoring tool and updated based on the user interaction. The Association process updates and stores its information in "Semantic Matrix", which is associated with each tracked surface. This matrix holds *state, permissions, properties, and relationships*, allowing interaction and view to work together.

State holds the number of stored interaction states and its interaction space (physical or information space). State allows the smooth interaction transition and assigns multiple UI elements based on the data exploration level. *Permissions* define which property is activated depending on the current user's context, and the current state. *Properties* are a set of attributes for instantiation of virtual information associated with this region. The following are currently eight different attributes stored in the properties set:

- P1: State: the current state's key.

- P2: Region Type: the location parameters and relative position to the region or the physical object, such as right edge, center or top.
- P3: Attached Data: a reference to data required for the UI control.
- P4: Occlude Condition: a Boolean value indicating if the UI control occlusion is allowable.
- P5: Physical Context: physical information presented in the region that is communicated to the UI control (the inherited physical context).
- P6: Interaction: the associated interaction with the control, which may be different to the state's invoking interaction.
- P7: Controls: the form of interaction that will be supported for the control.
- P8: Visualization: the type of visual element that will be presented.

Relationships are used to store the id of the connected regions, and to assign the weight of these connections, which is critical for the synergy between the virtual and physical elements. For instance, if a user clicked on the nutrition table on the product's box, a high-level description can be presented with virtual arrows pointing to more detailed information printed on the package.

B. Blended Controllers

The aim of the blended controls is to allow users to 1) view meaningfully fashioned, abstract data with their relationships, and 2) apply operations such as select, zoom, search, filter, and analyze. Using these techniques in AR requires a design adaptation to allow interacting with the physical objects and their associated information [2]. Our proposed techniques were drawn from the following paradigms for AR: Tangible User Interfaces (TUI), Adaptive User Interfaces (AUI), and Natural User Interfaces (NUI). In this section, we present a set of physical space interactions that can be used for blended controls: selection as a discrete gesture, pinching as a continuous gesture, proxemics as a user's perspective gesture, collision as physical objects' perspective gesture, and location-based as situational awareness.

The selection is a discrete control enables users to select/deselect physical objects, a region on the physical object, or a spatial point on the physical object. *Object selection* allows interaction with one or more physical objects from the real scene (Figure 3a). *Region selection* interacts with the regions on each physical object (Figure 3b). Based on the state, the act of picking up and holding the object transitions between different types of interactions. The *clicking* (touching the object at a point) interacts with spatial points on the physical object.

The pinch gesture is a continuous control that can provide a numerical values and vector director (see Figure 3c), which can be useful zooming and sliding.

Proxemic is an interaction based on the user's view, by calculating the distance between the user's view camera and the tactile physical objects. Proxemic provide intuitive interaction with the physical space. For instance, by moving the object nearer and further to the camera view (see Figure 3d), the amount of data presented changes, where holding a physical object and bringing it nearer the camera view can reflect an interest value in the object.

Collision is an interaction based on the spatial relationship of multiple objects to provide information pertinent to the objects' combination. For instance, users can compare or accumulate the information associated with the physical objects, by putting them side-by-side.

Location-based interaction is another multiple-objects interaction, based on the spatial location of the objects in the scene, independently (not combined as in the collision interaction). It is used for assigning priority values, or sorting based on the physical objects' location.

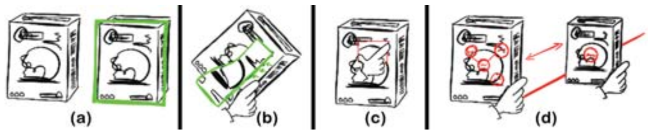


Fig. 3: Physical space interaction techniques. (a) Object Selection. (b) Region Selection. (c) Pinch gesture. (d) Proximic interaction.

C. Blended Views

The blended views hold the GUI elements and responsible for generating the blended widgets. The uniqueness of our blended views is attaching the widgets and the visuals based on the model's semantics matrix to leverage the meaning of the physical context. The semantic fusion of the UI elements to the physical world allows physical objects to be part of the interactive information process. In this section, we present a set of example UI elements that work in concert with the controls to achieve the blending aim.

AR information visualization needs a number of controls to allow users to manipulate different types of data, such as nominal, Boolean, and hierarchical. In this paper, we propose a set of situated UI elements that can be used for SA (Figure 4). These elements are used to handle different data types, such as menus for hierarchical data, slider bars for nominal values, and toggle buttons for Boolean values. All the proposed controls are designed for dynamic appearance creation for the Semantic Matrix.

Blended menus change their appearance and items based on the physical context calculated by the semantic matrix. We present four situational menus for our blended model: *dynamic, mapped, off-objects* and *drop-down*. The transition between these menus is based on the state and the information stored in the semantic matrix. *Dynamic* menus can be dragged and relocated to any place on the physical objects (see Figure 4a), with dynamic size, shape, and color of the menu based on the physical context. These menus use the regions' meta-values to restrict the location of the menu based on *Occlusion's* values stored in the semantic matrix. *Mapped* menus are statically located based on the *Region Matrix* in the *Model component* (see Figure 4b), which is easily recognized. *Off-objects* menus are used to align the menus outside the physical object (see

Figure 4c), not to mask the physical object with the menu items. *Dropdown* menus are fixed location, and its size is calculated based on the physical object's size (see Figure 4d).

Sliders are used to assign a numerical value and a vector direction. We present two type of sliders: one dimensional and two dimensional. The *one-dimensional* slider assigns nominal values (see Figure 4e). The *two-dimensional* slider bar assigns two values: the first is the horizontal displacement; and the second is the vertical one, assigning area and vector direction.

Toggle controls are assigned to regions or spatial points on the physical object to represent a boolean value which can be used for filtering and analyzing operations.

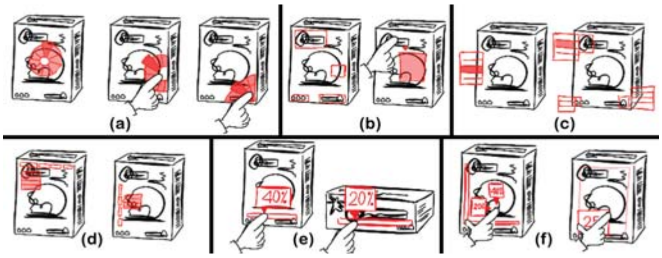


Fig. 4: User Interface controls. (a) Dynamic Menu. (b) Mapped Menu. (c) Off-object Menu. (d) Dropdown Menu. (e) One-dimensional slider bar. (f) Two-dimensional slider bar.

V. Example Of Blended User Interface

In this section, we present an interactive blended interaction session, based on combinations of the interaction techniques and UI elements that are controlled by the semantic matrix. The semantic matrix parameters were stored using an authoring tool to manually store model parameters to an external file for persistent storage employed by the application. This authoring process can be extended with image processing or user based authoring through a community developed content. Our example serves as a proof-of-concept to demonstrate the benefits of our proposed system.

Fig. 5: (a) Mapped a product box by using Vuforia virtual buttons. (b) The heatmap shows the regions' relationship. (c) Relationships change based on region id in the Semantic Matrix.

We developed the proposed example using Unity and Vuforia for feature tracking. We used Vuforia virtual buttons to divide the object's surface into a grid of tracked regions (Figure 5a), associated with the semantic matrix that was stored using the mentioned authoring tool. We applied a similar occlusion based approach that Lee, Billinghurst, and Kim employed for interaction with ARToolkit markers [17]. Kim, Widjojo, and Hwang extended this concept for more precise selection [15], and we will investigate this improved technique in the future. One of our main contributions is attaching interactive regions to our Semantic Matrix. Figure 5b and 5c shows heat map visualization for the relationship between the box's regions.

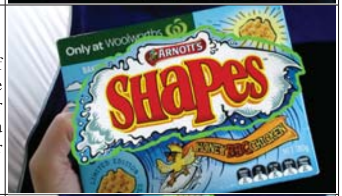
State 1: *Select a physical object* The user moves the AR display with a camera to scan products on the shelf and selecting the product. The selected product is highlighted by a green frame.	
State 2: *Explore and select a region* The user then takes one of the products off the shelf, as they are interested in more detailed information about this particular product. This user's interaction will invoke a detailed view of the product which the user is holding, enabling region selection.	
State 3: *Interact with contextual regions* When the user clicks on the box, the blended system will activate the holding box as a control input and highlighting the active regions of interest as toggle buttons. These regions of interest are driven by the contextual features of the box.	
State 4: *Menu manipulation* The user then clicks on one of the toggle buttons, invoking a dynamic menu. The user drags the menu over the product box, leading to altering the menu items based on the contextual data point beneath the menu. The menu's appearance changes based on the constraints stored in the semantic matrix.	
State 5: *Pinch Zoom* When the user starts to interact with two touch fingers on the box surface, the interaction control changes to a magnifying pinch zoom.	
State 6: *Analyze* The user starts to interact with multiple products; the system provides analyzing operations. The user put the products side-by-side to visualize the combined nutrition values of multiple products.	

TABLE I: Blended Interaction Session

In this section we are fusing the blended control components in concert to employ zoom, selection, compare, Details-on-Demand, and analyze operations. Table 1 depicts what the user would see during a series of interaction states in the blended space. The user moves between the states based on the predefined parameters of the semantics matrix, defining the invoking trigger for each state, permissions and parameters associated with the mapped contextual feature. The remainder of the section details how our Blended UI Controls model supports these forms of interactions.

A. State 1: Select a physical object

Physical selection allows users to select one or more physical objects from the real scene. Table 2 shows the semantic matrix stored values used to activate the objects selection. The state condition (state) is assigned to invoke this state when the proxemic distance is far distance. The near and far values are calculated based on a threshold value, which was assigned through data authoring. The physical selection (State 1) is applied to all regions P1 with associated highlight frame for the visualization element. The size of this frame is generated based on the stored dimension of the real object and is assigned the green color P7. P3 disallows occlusion to the entire physical box.

P0: *State*: Proxemic hitValue =far & Selection.click = false
P1: *Region type*: ALL
P2: *Attached data*: NULL
P3: *Occlude condition*: False
P4: *Physical context*: NULL
P5: *Interaction*: _fullObject, Proximic
P6: *Controls*: NULL
P7: *Visualization*: VirtualBlending.Highlight(objectDim,Color.green)

TABLE II: State 1 semantic matrix (select a physical object).

B. State 2: Explore and Select a region

State 2 invokes when the user holds one of the products and brings it near to the viewer's camera. Table 3 shows the different parameters between state 2 and state 1, highlighted in yellow. The table shows that this state invokes when the proximal distance is near (P0). This state uses the product box's texture to represent the information, by pop-up to highlight the selected region (P7), with occlusion permission (P3). The user can tilt and move the objects to select a region on the box using a ray tracing intersection.

P0: *State*: Proxemic hitValue =near & Selection.click = false
P1: *Region type*: ALL
P2: *Attached data*: _regionTexture
P3: *Occlude condition*: True
P4: *Physical context*: NULL
P5: *Interaction*: _partObject, Proximic
P6: *Controls*: NULL
P7: *Visualization*: RealBlending

TABLE III: State 2 semantic matrix (explore and select a region).

C. State 3: Interact with contextual regions

State 3 enables the conversion of the product box to an interactive surface, allowing users to click on the physical printed context to invoke GUIs such as menus and sliders. Table 4 lists the state parameters that are being stored in the semantic matrix. This interaction state is invoked when the object's proxemic distance is near, and the user has clicked on the product box (S0). The controllers in this state are depended on the contextual features of the real scene (P4), allowing the user to interact with the physical box by a clicking gesture (P1), and using a green highlight as visual affordance (P7).

P0: *State*: Proxemic hitValue =near & Selection.click = true
P1: *Region type*: Array SemanticMatrix.getRegions(_St3_active)
P2: *Attached data*: _regionTexture
P3: *Occlude condition*: True
P4: *Physical context*: _regionTexture
P5: *Interaction*: Select= hl_click, Proximic.
P6: *Controls*: NULL
P7: *Visualization*: VirtualBlending.Highlight(objectDim,Color.green)

TABLE IV: State 3 semantic matrix (explore and select a region).

D. State 4: Menu manipulation

The menu manipulation state allows the user to drag the menu over the different contextual feature on the physical box to explore a detailed breakdown, such as nutrition items, on the nutrition table region, or different flavors on the flavor picture. The blended menu appearance changes based on the region properties, where the top regions invoke a drop down menu, the side regions invoke the off-object menu, and the rest of the regions invoke the circular menu. The different menus are specified in P6, and the semantic matrix holds the equivalent menu for each region type in SematicMatrix.controlType, as specified during the authoring. The shapes of these menus dynamically change based on the user interaction, the physical context, and semantic matrix values (P6, P7). Table 5 shows the state parameters, highlighting the different ones in yellow. The table shows that this interaction is invoked when the interaction type change from click to drag (P5).

P0: *State*: Proxemic hitValue=near & Selection.click = true
P1: *Region type*: Array SemanticMatrix.getRegions(_St3_active)
P2: *Attached data*: _regionTexture
P3: *Occlude condition*: True
P4: *Physical context*: _regionTexture
P5: *Interaction*: Select= _drag, Proximic.
P6: *Controls*: menu.Type(SematicMatrix.controlType(region_id))
P7: *Visualization*: SemanticMatrix.Visual(region_id, control_id)

TABLE V: State 4 semantic matrix (menu manipulation).

E. State 5: Pinch Zoom

The pinch zoom operation allows users to enlarge the physical surface's information. This requires both the physical object and virtual presentation spaces to operate synchronously, allowing users to see small printing on a product box or a small picture attached to a magazine article. Table 6 shows the semantic matrix parameters of the pinch zoom, with two fingers interaction invoking parameter (P0).

P0: *State*: Proxemic hitValue=near & Selection.click = true & touches=2
P1: *Region type*: ALL
P2: *Attached data*: _regionTexture
P3: *Occlude condition*: True
P4: *Physical context*: NULL
P5: *Interaction*: _pinch
P6: *Controls*: NULL
P7: *Visualization*: RealBlending

TABLE VI: State 5 semantic matrix (Pinch Zoom).

F. State 6: Analyze

Analyze state provides a physical comparison operation, as the user hold an object and by putting the objects side-by-side, they can combine the nutrition information, or compare, or sort them. Tables 7 show semantic matrix's parameters for an aggregation task, enabling users to accumulate the total nutrition budget for multiple products. This state is invoked by a collision interaction (P0, P5), and the displayed annotation is a calculation based on star nutrition [12] stored functions.

P0: *State*: Selection.click = true & collision=true
P1: *Region type*: ALL
P2: *Attached data*: NULL
P3: *Occlude condition*: True
P4: *Physical context*: NULL
P5: *Interaction*: Collision
P6: *Controls*: NULL
P7: *Visualization*: VirtualBlending.Annotation(nutritionFun(starPoints))

TABLE VII: State 6 semantic matrix (Analyze).

G. Conclusion

This paper presents a model for SA Blended User Interface Controls, as a step forward for interactive visualizations in AR. We introduce the model framework and examples using this model. This model is associated with a Semantic Matrix allowing interactive AR information to work in synergy with contextual information on the physical artifact. The proposed techniques were deployed on different displays. Our presented model is contextual and state awareness, solving the situated analytics dual interaction space challenge (physical and interaction). Our proposed solution uses the physical context affordance for achieving auto-transition state model and to enhance information understanding.

REFERENCES

[1] R. Azuma and C. Furmanski. Evaluating label placement for augmented reality view management. In *Proceedings of the 2nd IEEE/ACM international Symposium on Mixed and Augmented Reality*, page 66. IEEE Computer Society, 2003.

[2] R. T. Azuma. A survey of augmented reality. *Presence: Teleoperators and virtual environments*, 6(4):355–385, 1997.

[3] M. Baglioni, E. Lecolinet, and Y. Guiard. Jerktilts: using accelerometers for eight-choice selection on mobile devices. In *Proceedings of the 13th international conference on multimodal interfaces*, pages 121–128. ACM, 2011.

[4] B. Bell, S. Feiner, and T. Höllerer. View management for virtual and augmented reality. In *Proceedings of the 14th annual ACM symposium on User interface software and technology*, pages 101–110. ACM, 2001.

[5] D. Benyon, O. Mival, and S. Ayan. Designing blended spaces. In *Proceedings of the 26th Annual BCS Interaction Specialist Group Conference on People and Computers*, pages 398–403. British Computer Society, 2012.

[6] 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*, pages 73–80. ACM, 1993.

[7] T. Chandler, M. Cordeil, T. Czauderna, T. Dwyer, J. Glowacki, C. Goncu, M. Klapperstueck, K. Klein, K. Marriott, F. Schreiber, et al. Immersive analytics. In *Big Data Visual Analytics (BDVA), 2015*, pages 1–8. IEEE, 2015.

[8] N. ElSayed, B. Thomas, K. Marriott, J. Piantadosi, and R. Smith. Situated analytics. In *Big Data Visual Analytics (BDVA), 2015*, pages 1–8. IEEE, 2015.

[9] N. A. ElSayed, B. H. Thomas, K. Marriott, J. Piantadosi, and R. T. Smith. Situated analytics: Demonstrating immersive analytical tools with augmented reality. *Journal of Visual Languages & Computing*, 36:13–23, 2016.

[10] N. A. ElSayed, B. H. Thomas, R. T. Smith, K. Marriott, and J. Piantadosi. Using augmented reality to support situated analytics. *Virtual Reality (VR)*, 2015.

[11] S. Feiner, B. MacIntyre, T. Höllerer, and A. Webster. A touring machine: Prototyping 3d mobile augmented reality systems for exploring the urban environment. *Personal Technologies*, 1(4):208–217, 1997.

[12] L. M. Fischer, L. A. Sutherland, L. A. Kaley, T. A. Fox, C. M. Hasler, J. Nobel, M. A. Kantor, and J. Blumberg. Development and implementation of the guiding stars nutrition guidance program. *American Journal of Health Promotion*, 26(2):e55–e63, 2011.

[13] H. Ishii. Tangible bits: beyond pixels. In *Proceedings of the 2nd international conference on Tangible and embedded interaction*, pages xv–xxv. ACM, 2008.

[14] D. Kalkofen, E. Mendez, and D. Schmalstieg. Interactive focus and context visualization for augmented reality. In *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 1–10. IEEE Computer Society, 2007.

[15] H. Kim, E. A. Widjojo, and J.-I. Hwang. Dynamic hierarchical virtual button-based hand interaction for wearable ar. In *2015 IEEE Virtual Reality (VR)*, pages 207–208. IEEE, 2015.

[16] G. E. Krasner, S. T. Pope, et al. A description of the model-view-controller user interface paradigm in the smalltalk-80 system. *Journal of object oriented programming*, 1(3):26–49, 1988.

[17] G. A. Lee, M. Billinghurst, and G. J. Kim. Occlusion based interaction methods for tangible augmented reality environments. In *Proceedings of the 2004 ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry*, pages 419–426. ACM, 2004.

[18] D. Leithinger and M. Haller. Improving menu interaction for cluttered tabletop setups with user-drawn path menus. In *Horizontal Interactive Human-Computer Systems, 2007. TABLETOP'07. Second Annual IEEE International Workshop on*, pages 121–128. IEEE, 2007.

[19] J. Looser, M. Billinghurst, and A. Cockburn. Through the looking glass: the use of lenses as an interface tool for augmented reality interfaces. In *Proceedings of the 2nd international conference on Computer graphics and interactive techniques in Australasia and South East Asia*, pages 204–211. ACM, 2004.

[20] J. Looser, R. Grasset, and M. Billinghurst. A 3d flexible and tangible magic lens in augmented reality. In *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 1–4. IEEE Computer Society, 2007.

[21] N. Marquardt, R. Diaz-Marino, S. Boring, and S. Greenberg. The proximity toolkit: prototyping proxemic interactions in ubiquitous computing ecologies. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 315–326. ACM, 2011.

[22] I. Oakley and J. Park. Motion marking menus: An eyes-free approach to motion input for handheld devices. *International Journal of Human-Computer Studies*, 67(6):515–532, 2009.

[23] W. Piekarski and B. H. Thomas. Tinmith-hand: unified user interface technology for mobile outdoor augmented reality and indoor virtual reality. In *Virtual Reality, 2002. Proceedings. IEEE*, pages 287–288, 2002.

[24] W. Piekarski and B. H. Thomas. Augmented reality user interfaces and techniques for outdoor modelling. In *Proceedings of the 2003 symposium on Interactive 3D graphics*, pages 225–226. ACM, 2003.

[25] T. Piumsomboon, A. Clark, M. Billinghurst, and A. Cockburn. User-defined gestures for augmented reality. In *IFIP Conference on Human-Computer Interaction*, pages 282–299. Springer, 2013.

[26] D. Schmidt, J. Frohnhofen, S. Knebel, F. Meinel, M. Perchyk, J. Risch, J. Striebel, J. Wachtel, and P. Baudisch. Ergonomic interaction for touch floors. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 3879–3888. ACM, 2015.

[27] H. Slay, M. Phillips, R. Vernik, and B. Thomas. Interaction modes for augmented reality visualization. In *Proceedings of the 2001 Asia-Pacific symposium on Information visualisation-Volume 9*, pages 71–75. Australian Computer Society, Inc., 2001.

[28] B. Thomas, K. Grimmer, J. Zucco, and S. Milanese. Where does the mouse go? an investigation into the placement of a body-attached touchpad mouse for wearable computers. *Personal Ubiquitous Comput.*, 6(2):97–112, Jan. 2002.

[29] B. H. Thomas, G. S. Von Itzstein, R. Vernik, S. Porter, M. R. Marner, R. T. Smith, M. Broecker, B. Close, S. Walker, S. Pickersgill, et al. Spatial augmented reality support for design of complex physical environments. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*, pages 588–593. IEEE, 2011.

[30] E. Veas, R. Grasset, E. Kruijff, and D. Schmalstieg. Extended overview techniques for outdoor augmented reality. *IEEE transactions on visualization and computer graphics*, 18(4):565–572, 2012.

[31] S. Von Itzstein, B. H. Thomas, R. T. Smith, and S. Walker. Using spatial augmented reality for appliance design. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*, pages 316–318. IEEE, 2011.

[32] J. A. Walsh, S. von Itzstein, and B. H. Thomas. Ephemeral interaction using everyday objects. In *Proceedings of the Fifteenth Australasian User Interface Conference-Volume 150*, pages 29–37. Australian Computer Society, Inc., 2014.

[33] S. White, D. Feng, and S. Feiner. Interaction and presentation techniques for shake menus in tangible augmented reality. In *Mixed and Augmented Reality, 2009. ISMAR 2009. 8th IEEE International Symposium on*, pages 39–48. IEEE, 2009.