

Research Article

Moritz Skowronski*, Daniel Klinkhammer, and Harald Reiterer

Argus Vision: A Tracking Tool for Exhibition Designers

<https://doi.org/10.1515/icom-2019-0001>

Abstract: Contemporary exhibitions are increasingly staged using extensive and often interactive media. To create such exhibitions, exhibition design companies employ professionals from a wide range of different disciplines. The support of interdisciplinary exhibition designers in the design process is one goal of research in Human-Computer Interaction. This includes the deployment of Do-It-Yourself (DIY) Tools that enable professionals from all disciplines involved to design and create interactive media themselves. In this paper, we will present Argus Vision, a DIY Tool, which allows exhibition designers the use of camera-tracking to rapidly prototype and develop immersive exhibitions and interactive installations. We successfully used Argus Vision in two real-world case studies both in the prototyping and in the deployment of two installations in exhibitions. Additionally, we conducted expert interviews with exhibition designers, investigating the tool's usefulness for them.

Keywords: Exhibition Design, Tracking Tool, Rapid Prototyping, DIY, Kinect, Blended Museum, Proxemics

1 Introduction

In contemporary exhibitions, artifacts are no longer solely presented in a clean exhibition room but are staged in order to create a thematically coherent narrative for visitors [13]. In this approach, called scenography, a variety of different instruments is used, such as such as light, sound, graphic design, and interactive media [9]. This vast number of techniques makes it necessary for exhibition designers to work in interdisciplinary teams or to outsource the implementation of parts of the exhibition [20]. Especially the realization of interactive exhibitions is very expensive,

*Corresponding author: **Moritz Skowronski**, Human-Computer Interaction Group, University of Konstanz, Konstanz, Germany, e-mail: moritz.skowronski@uni-konstanz.de

Daniel Klinkhammer, Harald Reiterer, Human-Computer Interaction Group, University of Konstanz, Konstanz, Germany, e-mails: daniel.klinkhammer@uni-konstanz.de, harald.reiterer@uni-konstanz.de

allowing only large exhibition design companies to design and deploy such exhibitions. One approach in recent research in Human-Computer Interaction (HCI) is to provide Do-It-Yourself (DIY) Tools to enable exhibition designers from all disciplines and with varying programming skills to implement interactive exhibitions themselves [14, 16, 19]. The need for such tools is highlighted by the example of MESO Digital Interiors, a leading exhibition design company that provides the DIY Software Tool vvvv,¹ a visual programming environment, which is widely used for the rapid prototyping and development of interactive media and explicitly made for people with basic programming skills. By conducting five semi-structured interviews with exhibition designers of leading exhibition design companies in Germany we elicited requirements for the design of such DIY Tools. They have to be easy to use and therefore reduce the required programming skill to a minimum, provide direct visual feedback on interaction, and should support rapid prototyping and the fast changing of settings on site. Furthermore, these tools should be designed explicitly for common interactive use cases in exhibitions. These include interactive walls [17], tabletops [4], tangible interaction [2] or kinetic installations [1], all of which are research areas in HCI and therefore well explored.

A new trend in many modern exhibitions is to immerse the visitors, casting them as actors instead of mere spectators. In such exhibitions, visitors have an actual influence on parts of the exhibition explicitly by touching the exhibits or by body movement or implicitly by entering a specific area in the exhibition [12, 6]. We can also explain this increasing interest in such exhibition concepts by the spread of ubiquitous computing (ubicomp) technologies. The term was coined in 1991 by Mark Weiser and describes his vision of a future, where technology seamlessly integrates into everyday life [25]. With devices such as tablets and smartphones that many people carry with them every day, part of this vision has become a reality. Weiser also describes the communication between all these devices and the automatic adaptation of these devices to user situations. For example, a smartphone could automatically

¹ <https://vvvv.org/> (last access: January 7, 2019).

mute itself when the owner is in a concert or cinema. We can transfer this vision to immersive exhibitions. For example, an exhibit could only show additional information if a visitor turns to it or stays there for a particular time; room lighting could change depending on the number of people in the room. These immersive interaction concepts are by and large implemented using implicit and explicit proxemic interactions, which are difficult to design [15].

Our main contribution is the provision of the DIY Tracking Tool Argus Vision. It allows exhibition designers with basic or no programming skills the design and implementation of implicit and explicit proxemic interactions by abstracting complex tracking algorithms using multiple Kinect version 2 cameras into a simple user interface. We support the rapid prototyping and development of proxemic interaction concepts and the change of settings on site by the use of Triggerzones, virtual three-dimensional areas, which detect the presence of visitors. To evaluate whether the tool is applicable in exhibitions and usable by exhibition designers, we used the tool in the design and deployment of two installations at exhibitions and conducted five expert interviews with exhibition designers.

2 Background and Related Work

We surveyed contemporary exhibitions and their use of tracking technologies to gain an understanding of which features a Tracking Tool for exhibition designers has to provide. Based on these findings, we investigated existing tracking toolkits and their applicability for the exhibition design sector.

2.1 Exhibitions Using Tracking Technologies

The implicit and explicit interaction with exhibits and installations through natural body movement plays an integral role in the creation of immersive exhibitions [10]. In order to detect the movement of visitors in an exhibition space, different tracking technologies have been used such as depth cameras [23], RFID [5], or infrared sensors [18]. The choice of tracking technology strongly depends on which data an exhibit requires. We distinguish between three general use cases for interactive media using tracking technologies in museums.

- Interactive media using gesture interaction
- Interactive media using full-body interaction
- Interactive media using proxemic data

In the following, we will describe these use cases by highlighting examples from contemporary exhibitions and their use of tracking data.

2.1.1 Interactive Media Using Gesture Interaction

Gesture interaction is an interaction technique rarely used in exhibitions. One reason for that is that even state of the art controllers, such as the leap motion, still require a rather controlled environment to function flawlessly. Moreover, learning different gestures to control an installation poses an additional challenge for the visitors. In contrast, simple tangible installations can be found often. While these installations are mostly marker-based or using capacitive sensors, they can also be implemented using infrared sensors or camera tracking. A simple use case for such an installation can be seen in the exhibition *Salt Worldwide* [2]. Here, visitors can touch capacitive salt crystals, which are mounted on a table showing a large world map. When a crystal is touched, additional information is shown about the salt mines at their respective location.

2.1.2 Interactive Media Using Full-Body Interaction

To capture the whole body of visitors without using motion capture suits, depth cameras like the Kinect can be used. At an interactive installation in the exhibition *Micropia*, the complete skeleton of the visitors is tracked (see Figure 1) [3]. Visitors can explore a visual representation of their own body, which is shown on a large display. The display also shows additional information about specific body parts and the microbes living there. Other applications only use the contour of tracked persons. These installations, such as the installation *ICE*, offer a playful interaction [11]. Here, the visitors' contour and movement are represented abstractly in a projection (see Figure 1).

2.1.3 Interactive Media Using Proxemic Data

Most installations and exhibitions using tracking technology use proxemic data, which can be categorized based on five proxemic dimensions as described by Greenberg et al. – distance, orientation, movement, identity, and location [7]. In the majority of these immersive experiences, however, only the proxemic dimensions distance and location are used, since only the position of the visitors within a virtual interaction space or their entering of a particular area of the interaction space needs to be detected.

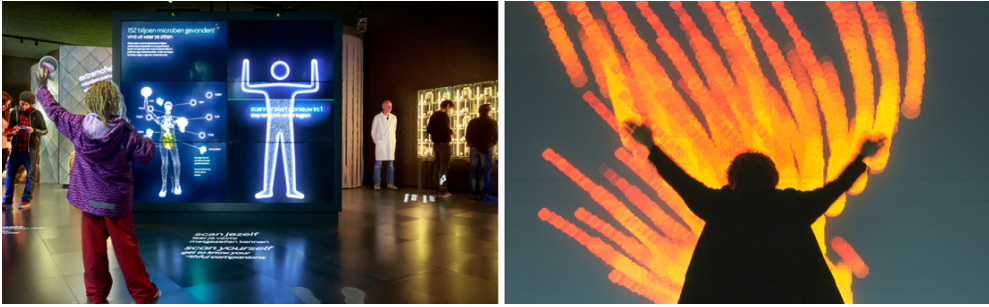


Figure 1: Two interactive installations using full-body interaction. Left: Micropia (Photo: Thijs Wolzak, © ART+COM AG). Right: ICE (Photo: © Jun Takagi).



Figure 2: Two installations where the position of the visitors influences either a small interaction space (left) or the whole exhibition space (right). Left: Interactive (Photo: © Electroland). Right: Time Machine (Photo: © TAMSCHICK).

This data is often used to provide the visitors with additional content when they enter a specific space around an exhibit. For example, in the exhibition *Mariko Mori – Oneness*, circles were drawn on the floor in front of individual works of art [12]. Infrared sensors detected if a visitor entered such a circle. The visitors were then able to hear additional information about the artwork in front of them from directional loudspeakers. In a media station, Cafaro et al. combined a depth camera and RFID technology to present personalized content to several visitors simultaneously [5]. RFID was used to determine the identity of the visitors, the depth camera to determine their distance to the media station. In the installation *Interactive*, visitors are tracked on a large floor of LED lights [6]. The position of the visitors determines the displayed light patterns (see Figure 2).

In the immersive exhibition *Time Machine* both the walls and the floor in the room are staged using projections, which tell a continuous story (see Figure 2). In order to give visitors the feeling of being part of the story, their position and movements are reflected abstractly on the floor [23]. Lastly, interactive media can also be used to create a narrative between different exhibits, connecting them either by presenting information that thematically connects different exhibits or by guiding the visitors along

a specific path through the exhibition. In the showroom of the company *KUKA*, visitors are registered by a tracking system, which controls a light that guides them through the exhibition [24]. However, in this area only a few examples for the use of interactive media using tracking technology exist.

2.2 Tracking Toolkits

In the previous section, we have presented different use cases of tracking technology in museums. While some examples for the use of gesture or full-body interaction exist, a majority of installations only make use of proxemic data. Therefore, in the design of *Argus Vision*, we focused on enabling the design of these installations.

Several toolkits have been developed with the aim to provide easier access to proxemic data using tracking devices, albeit with different target user groups. The *Proximity Toolkit* [15] and *SoD-Toolkit* [21] are toolkits for the prototyping and development of ubiquitous spaces. Both toolkits feature GUIs to monitor the tracking results and set up the devices. However, these tools were neither designed for the use in exhibitions nor for people with basic programming skills. A major disadvantage of the *Proxim-*

ity Toolkit is its dependence on motion capture systems to use its full set of features, e. g., the use of fixed-features to detect whether a person is entering a specific area in the room. As these systems need markers to detect persons, they are not suitable for the use in exhibitions. The Kinect's skeleton tracking can be used to enrich the tracking data of the motion capture system. The SoD-Toolkit allows the use of a variety of sensors like mobile devices to detect the orientation of a person or the Leap Motion for gesture detection. To detect the position of people, it uses the skeleton tracking of the Kinect. Since both toolkits rely on the proprietary skeleton tracking of the Kinect, it is necessary to position the camera in body height and parallel to the floor, which creates difficulties in hiding the hardware in an exhibition. Additionally, both toolkits provide a range of data and settings for research in proxemic interaction. A majority of these is not necessary for the use in exhibitions and only makes the user interface and the processing of tracking data more complex.

In contrast, TSPS² and KinectA [8] are tools designed explicitly for the use in creative applications and exhibitions. Both feature simple user interfaces and allow the use of blob or skeleton tracking algorithms to detect objects and persons in the camera image. However, they do not offer support for multiple cameras, thus limiting the exhibition space that can be effectively monitored. Most importantly, they only provide access to continuous tracking data, which still needs to be processed further (e. g., calculating the position of a blob in relation to an installation) which is a challenging step for people with little programming skills. Thus, even though these tools are developed for the use in exhibitions, they are still intended for designers with a background in programming.

In contrast to the related work, we provide an easy and fast way for exhibition designers to use camera-tracking that is still complex enough to implement a wide range of contemporary installations and interactive exhibits. At the same time, we support the rapid prototyping of such me-

dia remotely and on site. We primarily focused on enabling designers with few or no programming skills to make use of the earlier mentioned continuous and binary measurements of the proxemic dimensions distance and location.

3 Argus Vision

In this section, we will introduce the DIY Tracking Tool Argus Vision. First, we will explain the architecture and user interface of the tool to show how the tool can be used in exhibition spaces. Then, we will describe the user detection and tracking algorithm used in the tool, before we introduce the Triggerzone functionality. We specifically highlight their benefit for the usability of the tool. The source code of Argus Vision is available to the public³ and written in Processing,⁴ a programming environment and language commonly used for creating interactive media. This allows exhibition designers to participate in the further development of the tool.

3.1 Overview

Argus Vision can be regarded as middleware, providing all the necessary tracking data to be used in animations, installations, and other interactive media. Argus Vision itself is divided into two main components (see Figure 3), which are communicating via local network – Argus Control and Argus Kinect.

The main task of Argus Kinect is to process the raw depth data from the Kinect and to send the processed data to Argus Control. When the application starts, it automatically starts the Kinect and loads settings from the previous session if available. Since all settings are made in Argus Control, the user interface of Argus Kinect only allows the user to enter the data required for communication with

² <http://www.tsps.cc> (last access: January 7, 2019).

³ <https://github.com/hcigroupkonstanz/ArgusVision> (last access: January 7, 2019).

⁴ <https://processing.org/> (last access: January 7, 2019).



Figure 3: The architecture of Argus Vision.

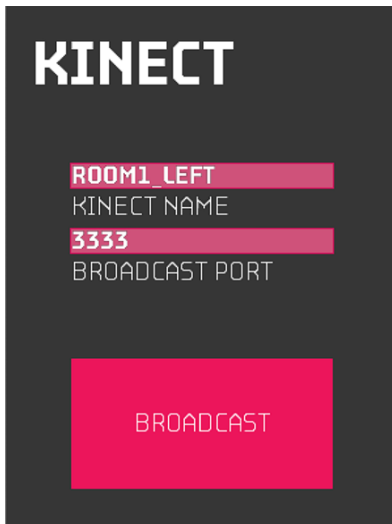


Figure 4: The user interface of Argus Kinect.

Argus Control (see Figure 4). These are a unique name for the Kinect and a port on which Argus Control and Argus Kinect will communicate. This division makes it possible to use multiple Kinects, each controlled by one computer running an entity of Argus Kinect. Furthermore, it ensures that for every Kinect the tracking algorithm performs at the same speed, which is especially important in applications where tracking data should be visually represented without glitches, e. g., if the tracked silhouette is shown in a projection. Argus Kinect forwards the processed tracking data to Argus Control, which bundles the data from all Argus Kinect-entities and sends these results to one client, for example, a programming environment for interactive media (vvvv, Processing), a game engine (Unity) or DIY Hardware Tools (Arduino).

In order to make the setup of a connection between Argus Control and Argus Kinect easier, we implemented a broadcasting method, automatically connecting an Argus Kinect-entity to an available Argus Control-entity if they are in the same network. As long as Argus Kinect is not connected to Argus Control, it sends a communication request via local network in a specific time interval and waits for a response from Argus Control. If it receives a response, Argus Control and Argus Kinect automatically set up a TCP connection for the lossless transfer of settings and a UDP connection to stream images from the respective Kinects. The tracking data itself is sent via OSC, a popular and easy-to-use network protocol in interactive media. This ensures that many applications commonly used in exhibitions can receive and process the data.

All of the functions available in Argus Vision are bundled into the user interface of Argus Control. It is divided

into three parts: In the Communication and Triggerzone settings, the user can set up a connection between Argus Control and Argus Kinect as well as between Argus Control and a client. Moreover, one can manage the Triggerzones as seen in Figure 5a. The user can add them to or delete them from a scene and can position and rotate them using exact measurements as input. The tracking algorithm – specific for every Kinect-entity – is set up in the Kinect controls (see Figure 5b). The configuration consists of two steps. Firstly, the current camera frame is saved as a background image for the tracking. Secondly, the user has to set two threshold values, which determine when a recognized change in the camera image is registered as a person. In the Kinect view (see Figure 5c), the user has access to four different visualizations of the Kinect data.

A depth and infrared stream can be used to visualize the room regardless of the room’s lighting, while a tracking view shows all detected persons in the room. A point cloud view is used to manage the existing Triggerzones and to view the room in a 3D environment. By using all of the functions of Argus Vision, users have access to the following data:

- For each person: id, age (time of stay in the scene, measured in frames), centroid, contour, velocity, acceleration
- For each Triggerzone: id, occupancy level, occupancy per person

3.2 User Detection & Tracking

In Argus Vision, we used a modified form of the background subtraction algorithm, where the current frame of a camera is compared to a previously saved background frame in order to find differences between the two. In most cases, this process is applied to 2D images, where the color values of each pixel have to be compared to the color values of the pixels of the background image (e. g., [26]). The pixels of the current frame which exceed a certain difference threshold are written to a new image, which then only shows those parts of the image that have changed. In Argus Kinect, we can simplify this process by just comparing the depth value of each pixel in the current frame with the depth value of the same pixel in the background frame. If the difference between these values is larger than a user-specified threshold, we can assume that a feature in the room has changed, e. g., a person has moved or entered the field of view of the Kinect. We then color that pixel white, otherwise black. In Figure 6 we show two images resulting from this rule: Figure 6a is the result of applying the Background Subtraction algorithm to an empty

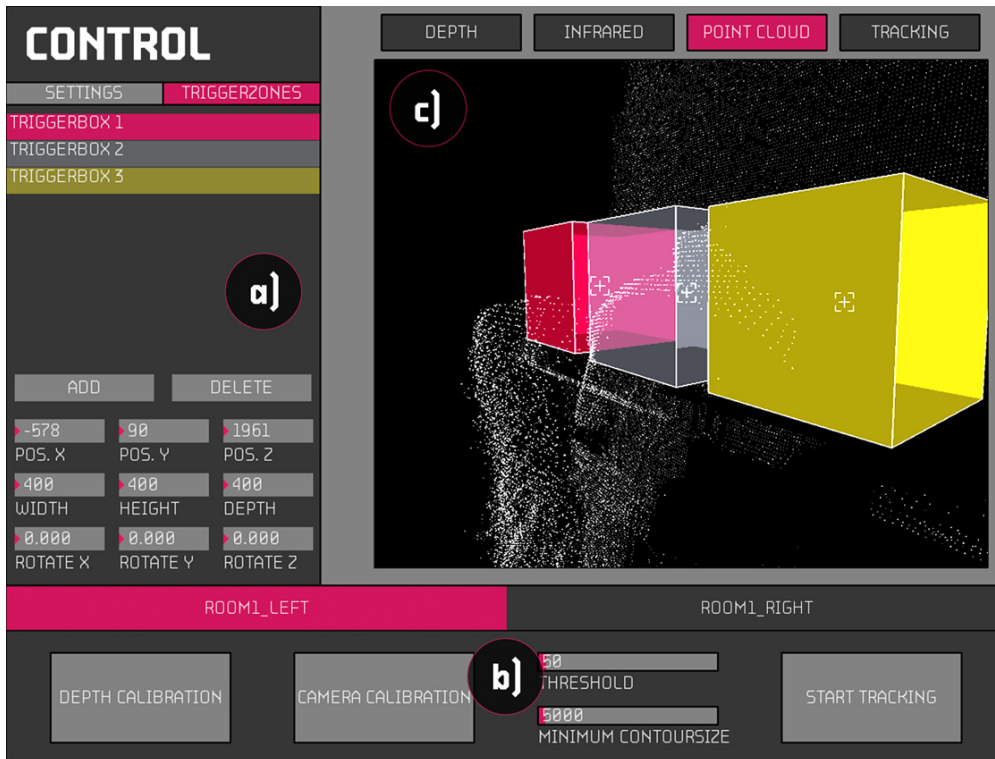


Figure 5: The user interface of Argus Control is divided into Communication and Triggerzone settings (a), Kinect controls (b) and the Kinect View (c). In the latter, three Triggerzones are placed on the edge of a table.

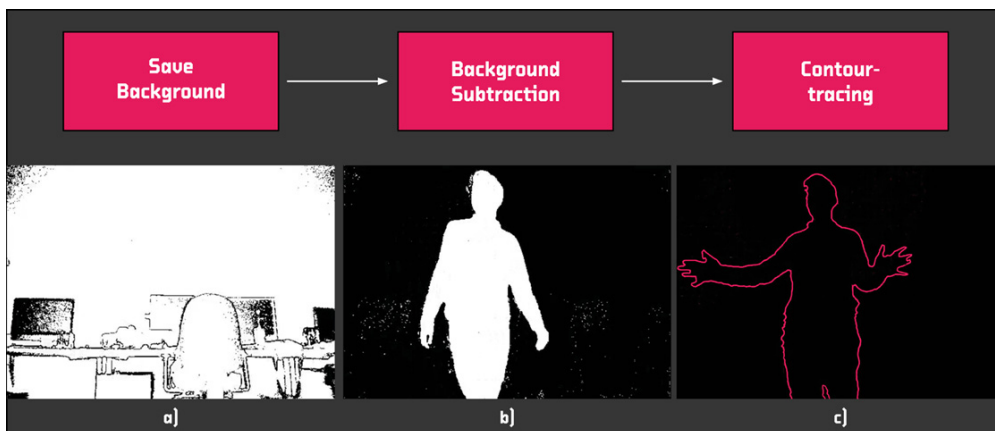


Figure 6: Results of the Detection Process of Argus Vision against an unsaved background (a), a saved background (b), and with activated contour-tracing (c).

background, a background where the depth value for each pixel is zero (white pixel) or undefined (black pixel). Figure 6b shows a successful background subtraction against the frame shown in Figure 6a. We then use OpenCV⁵ to extract additional features from this image, such as the con-

tour and centroid of each person (see Figure 6c). The result of this algorithm yields a finite set of 2D-contours, which in turn consist of a finite set of 2D-pixels. Since small errors (seen as smaller white pixels in figure 6b) can occur in the background subtraction, that can also be recognized as contours, we exclude contours that are smaller than a certain user-specified threshold. Finally, the pixels are assigned their depth values of the Kinect again. As a result of

⁵ <https://opencv.org> (last access: January 7, 2019).

this step, the 3D-contour of each person in the interaction space is obtained.

These centroids are then used to assign and maintain a constant id to each person as long as they remain in the camera image. For this, we calculate a distance matrix, comparing the distances of all centroids of the previous frame with those of the current frame. We then check this distance matrix for its minimum, i. e., we search for the smallest distance between a centroid from the previous and the current frame. If we find a minimum, we assign the id of the centroid of the previous frame to the centroid of the current frame. We then remove both centroids and all their entries from the distance matrix. We repeat this process until only new or old centroids are left. The former occurs exactly when one or more persons have entered the camera space, i. e., the total number of persons increased. We add the new centroids to the already tracked persons with a new ID. Conversely, this means that the latter case occurs exactly when one or more persons have left the camera space. We then remove these from the list of tracked persons.

This algorithm is a simple form of user detection and tracking and inferior to the quality of the skeleton tracking of the Kinect. However, the skeleton tracking only functions properly when the Kinect is mounted horizontally in front of the tracked persons. Yet in most exhibitions, the hardware is hidden to not alter the desired aesthetic of the respective exhibition room. Therefore, in contrast to most other GUI-toolkits for the Kinect, we needed to make it possible to use the Kinect from all different positions and angles, e. g., to mount it on the ceiling. Moreover, making use of skeleton data is not necessary for simple proxemic interactions, where mostly the dimension distance is used.

3.3 Triggerzones

With the introduction of Triggerzones, we greatly reduce the programming effort for exhibition designers. Triggerzones are virtual areas in a room, which detect the presence of visitors and therefore resemble light barriers in their behavior. At the time of writing, we support quad-shaped areas. However, since the algorithm used to detect whether a person is in the area can be used for every convex shape, we will support other shapes, especially radii, in a future version of the tool. The Triggerzones can be positioned and rotated freely in the camera-view using two different possibilities. Firstly, the users can use the settings menu to assign exact values for the position and rotation of each Triggerzone. However, positioning three-dimensional objects like the Triggerzones in two-

dimensional images can be difficult as it is hard to estimate the distance between the objects and the camera. In order to address this, we implemented a point cloud view. A point cloud shows three-dimensional points in a virtual three-dimensional space. Exhibition designers can navigate this view using a virtual camera with which they can view the room from all perspectives and drag and drop the Triggerzones to their desired position (see Figure 5c). This makes it easy to position the areas on object surfaces or floors accurately. With the Triggerzone concept, we reduce the results of the camera-tracking to on-and-off-statements, which can be easily processed in other applications through simple if-then-else expressions. In the simplest use case, this can mean that a sound or animation is triggered when a visitor enters an exhibition space. In a more complex example, several different-sized Triggerzones could be stacked to serve as proxemic zones around an exhibit. With this functionality, we support the rapid prototyping of interaction concepts by eliminating the need to write complex and time-consuming algorithms as well as the implementation of simple installations, e. g., triggering an animation when a person touches an exhibit or enters different areas in a room. Additionally, this enables exhibition designers to design their installations remotely and to adjust their interaction concepts to any other setting by repositioning, adding or deleting Triggerzones on site.

4 Evaluation

To evaluate whether Argus Vision is applicable in the area of exhibition design, we used the tool in the development of two installations in two different exhibitions. Both exhibitions were created as part of the research project and lecture series Blended Museum,⁶ in which students from the fields of architecture, history, communication design and human-computer interaction work on innovative exhibition concepts. The case studies differ not only thematically, but also in their use of Argus Vision. In the installation *Bruch*, we used the centroids of the detected persons. In the development phase of another installation, we used Triggerzones to detect the interaction of visitors. In addition, in this installation, we were able to examine to what extent Argus Vision could be linked to another DIY tool for the exhibition area. Lastly, we conducted five expert interviews with exhibition designers to fur-

⁶ <https://hci.uni-konstanz.de/en/research/research-projects/blended-museum/> (last access: January 7, 2019).

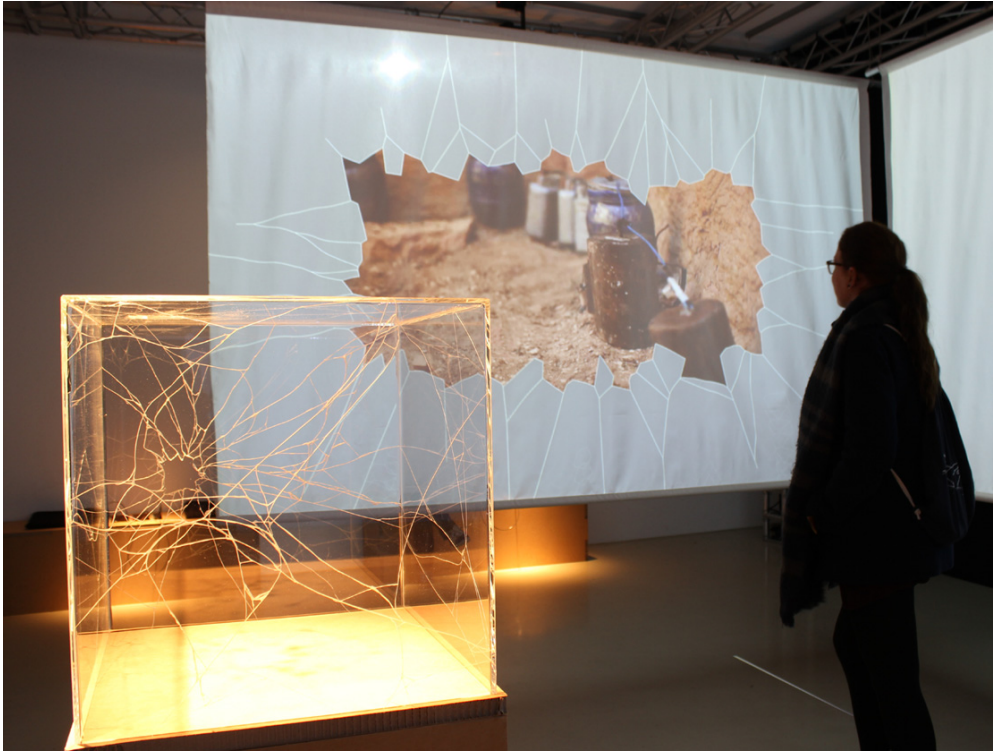


Figure 7: A visitor destroyed the virtual glass by getting too close to the projection.

ther assess the usefulness of the tool for the specific domain.

4.1 Case Study “Bruch”

We displayed the interactive installation *Bruch* as part of the exhibition *Tell Genderes – 20 Meter Menschheitsgeschichte*⁷ for six weeks at a local exhibition space. The installation was set up in a dedicated room, which focused on the current political situation in Syria and the associated loss of cultural heritage through art trade, theft, and destruction. A destroyed showcase was displayed in the middle of the room, metaphorically representing the destroyed and disappeared cultural artifacts. At a distance around the showcase, we hung gauze canvases from the ceiling. “Virtual glass” was projected on the canvases, mimicking the sides of the real showcase. Visitors were able to crack or destroy this virtual glass either accidentally or intentionally. The cracks in the glass were calculated using the position of the visitors in the room and were created precisely where the person stood in front of the canvas. The distance between the person and the canvas

determined the size of the crack. If a person crossed a certain distance threshold to the canvas, the glass collapsed entirely – in the resulting peephole images or videos of destroyed cultural heritage were shown (see Figure 7). After some time, the crack vanished, and the algorithm started again.

With Argus Vision, we were able to implement the glass-animation remotely, using the mouse and keyboard as a substitute for the actual presence of persons. We used the position of the mouse pointer to simulate the position of a visitor in front of the projection, whereas we used the keyboard to change the proximity of the visitor to the canvas. We integrated Argus Vision only on site and after we had set up the physical installation. We also wrote the animation itself in Processing. Thus, to receive and process the messages sent by Argus Vision to the animation application, we used the same OSC plugin and could rely on previously gained experience with the processing of the Argus Vision syntax. However, this process has to be performed in the same manner in every other tool or programming language. Therefore, we aim to encapsulate this process as a plug-in for a large number of different programming languages and tools. In this way, exhibition designers are provided easier access to Person and Triggerzone data without first having to perform a syntax analysis or create a suitable data structure.

⁷ <http://tell-genderes.de> (last access: January 7, 2019).

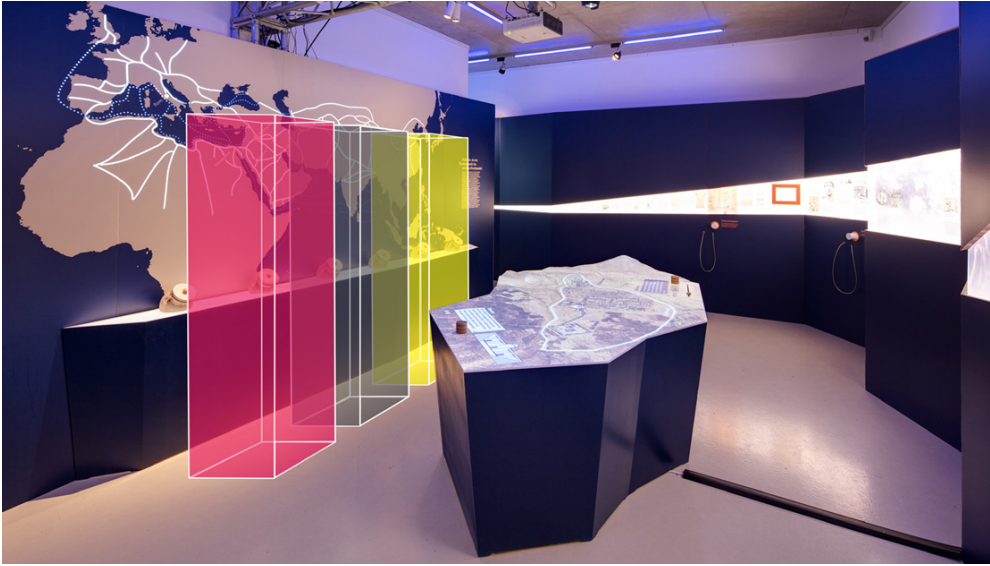


Figure 8: Three Triggerzones as we placed them in front of the interactive world map in the first prototype.

In the exhibition space itself, we mounted all hardware, including three Kinects on a truss on the ceiling facing the floor. To create an interactive, immersive experience for the visitors, we then merely substituted the input devices used in the prototyping phase with the centroid values from Argus Vision. The installation operated maintenance-free for the duration of the exhibition. During one week of the exhibition, we also observed how visitors interacted with the installation. Here, visitors initially walked towards the real showcase, thereby implicitly interacting with the canvases, which started to crack. Most visitors noticed then that they could interact with the canvases. It was particularly noticeable that many visitors engaged in playful interaction with the canvases, like hitting the canvases or jumping up and down in front of them. However, since we only used the position of the visitors for calculating the cracks, these interactions had no particular influence on the animation. Hence, we believe that most visitors either assumed the algorithm to be more elaborate than it actually was or that they did not care about the technology used as long as the animation seemingly reacted to their movements. This again underlines the applicability of Triggerzones, which we used in another case study.

4.2 Case Study “Rebuild Palmyra”

In the design and prototyping phase of the exhibition *Rebuild Palmyra?*, we used the Triggerzone functionality of Argus Vision [22]. Here, we implemented an interactive

world map that showed ancient trade routes and what resources were traded there. We printed the world map on a large wall over which we projected the respective trade routes. One of our first concepts was that the projection would change depending on where the visitors stood in front of the projection. Therefore, we divided that space into distinct areas. If a person entered one of these areas, the trade route associated with it would be shown. If the area was empty, the trade route disappeared. This is a typical scenario for the use of Triggerzones. We positioned same-sized Triggerzones on the floor in front of the projection and used these to control the display of the trade routes (see Figure 8). We then used the tool Resolume⁸ to change the content of the projection. Resolume is a VJ tool, which is widely used in interactive installations and events and allows among others the triggering and playback of a large number of video files at the same time. It also supports communication with other tools via OSC. We prerendered all content and set each file to be triggered on a specific OSC command sent by Argus Vision. This enabled us to build a functioning prototype in a short amount of time, with which we could test our intended interaction technique with a few subjects. Here, we could observe that some of the subjects did not understand the link between entering one of the areas and the display of trade routes. Because of this, we redesigned our interaction concept to a more explicit one: We placed amphoras in front of the world map, each having a different resource in it.

⁸ <https://resolume.com> (last access: January 7, 2019).



Figure 9: After lifting the lid of an amphora, a visitor watches the triggered projection.

When a visitor lifted the lid of an amphora, the respective trade route of that resource was displayed (see Figure 9). We again could test this principle by simply moving the Triggerzones to the position of the amphoras and reducing their size to fit around the amphoras. This use case highlights how the use of the Triggerzones can facilitate the fast and easy prototyping of interactive installations – especially combined with other tools.

4.3 Expert Interviews

In addition to the case studies, we conducted five expert interviews with exhibition designers from leading exhibition design companies. For this purpose, we contacted fifteen exhibition design companies and asked for an interview appointment. From this pool, we selected both offices with a stronger focus on architectural and design processes as well as offices with a stronger focus on the design of interactive installations. For the interview, we derived a set of ordered questions that served as a guideline during the interview and for the analysis. We divided the questions into three phases. Each interview started with a warm up-phase, in which we asked questions about current projects, current trends in exhibition design, and about work processes of the exhibition designers. After that, we asked questions about Argus Vision and its functions before we concluded with questions about possible applications and possibilities for further development. Since the exhibition designers spoke a lot about their own

projects and also based answers on questions about Argus Vision on experiences from these projects, anonymizing the interviews would have meant to leave out crucial information gained from the interviews. Therefore, the interviewees were allowed to review the transcriptions of the interviews and in turn permitted us to publish the interviews with their names. These transcripts were the basis of the analysis. Since large exhibition design companies typically outsource the implementation of interactive media [20], the exhibition designers did not use the tool themselves but were shown the functions and user interface of Argus Vision during the interview. We chose this approach because we anticipated quick and valuable feedback from experienced exhibition designers. In further research, we aim to provide the tool for smaller exhibition design companies to see how they apply the tool in real-world exhibitions. Since the experts did not use the tool themselves, we asked questions about the perceived usability of the tool, possible use cases and possibilities for further development. The interviewees were:

- Christoph Diederichs, Interaction Designer, Atelier Markgraph, Frankfurt am Main
- Dominik Hegemann, Associate, Atelier Brückner, Stuttgart
- Prof. Thomas Hundt, Founder and CEO, jangled nerves, Stuttgart
- Sebastian Oschatz, CEO, MESO Digital Interiors, Frankfurt am Main
- Prof. Eberhard Schlag, Project Director and Partner, Atelier Brückner, Stuttgart

All interviewees rated the user interface of Argus Vision positively and described it among other things as “slick” (all citations are translated by the authors) and “suitable.” (Diederichs) They also highlighted the clarity of the interface. Moreover, both the user interface as well as the programming logic were considered to be easily understandable by several respondents. Hundt, for example, described the interface as “tidy” and went on to explain that it gives the impression that both he and his employees could “get started with it” immediately. While Hege-mann stated that they would not use standardized tracking software, he saw a “large market [for Argus Vision] for simple installations, especially in temporary exhibitions and events.” In addition, several of the respondents could imagine that an exhibition design team could work with Argus Vision – “You do not have to be a computer scientist to be able to operate it, that is the most important thing.” (Schlag)

The interviewees also judged the Triggerzones functionality and its ease-of-use favorably. Three of the five interviewees could envision general applications for the use of Triggerzones – MESO was even developing an installation that used a similar functionality, according to Oschatz. Hundt also saw the possibility of using Triggerzones in the prototyping phase: Especially since the “[...] content and presentation comes first [...] it would, of course, be interesting if testing, programming and simulating such environments becomes easier and faster”. The experts also suggested possibilities for further development. These include the combination of multiple cameras to one virtual camera by blending the intersections of their field of view, the possibility to activate a skeleton or gesture tracking and the development of a frontend-editor, which allows the control of common interactive media like animations, lights, or sound in the user interface of Argus Control. One further aspect that was stressed by several interviewees was that installations in exhibitions need to be easily maintainable, both software and hardware wise. For this reason, Oschatz suggested that it would be beneficial for future exhibitions to use smaller standardized platforms such as the Raspberry Pi, because “if it doesn’t work, you throw away either the memory card or the PI, but you’re reproducibly back to how it worked before.” Moreover, all control inputs, such as the interface of Argus Control, should then be made available via a web browser to allow the remote control of all installations. Oschatz described Argus Vision as “one component in an ecosystem of different technologies used in exhibitions.” He continued that “if exhibition designers knew what they could do with it,” Argus Vision alone “enables sensational possibilities.” Con-

sequently, Oschatz does not regard the further development of Argus Vision as the most important future work, but “the strategic and long-term isolation of the components needed in the museum of the future” and thus, the deployment of similar DIY Tools for each of these components.

5 Results and Future Work

We demonstrated that Argus Vision can be used in exhibitions by outlining the successful use of the tool both in the prototyping phase as well as in the deployment of two installations in two separate exhibitions. The five expert interviews with exhibition designers gave us valuable feedback on the tool and its usefulness for the target group. We were able to gather ideas for the further development of Argus Vision. Overall, the interviewees judged the tool favorably and could envision that especially small exhibition design companies would use Argus Vision. In the future, we will provide the tool to exhibition designers of such exhibition design companies to evaluate the tool in a real-world setting. Furthermore, we believe that the concept of Triggerzones should be extended. The Triggerzones’ size and position are only set once, at the implementation of an installation. Additionally, the current version of Argus Vision only allows the use of quad-shaped areas, yet other shapes can be of interest to both exhibition designers and researchers. One example scenario would be the modeling of proxemic interactions using circle-shaped Triggerzones that follow the tracked users. In another scenario, one could imagine the size of the Triggerzones changing depending on the number of people in the vicinity. While these are interesting applications, the design space of such applications and their applicability in a real-world context still needs to be explored. Moreover, it should be noted that adding more complex functions to the tool directly contradicts our approach to making the tool easily usable for exhibition designers with no background in programming. We see Argus Vision as a module for using camera-tracking technology among a range of other technologies from which interactive installations are created. Therefore, we see our future work in line with the vision of one of our interviewees: We aim to investigate current exhibitions, elicit the technological components used and provide DIY Tools like Argus Vision for all of these different components. These tools should integrate seamlessly, making it possible to combine the functions of different DIY Tools to rapidly prototype and create interactive exhibitions.

6 Conclusion

Argus Vision is a DIY Tracking Tool that enables interdisciplinary exhibition designers to design and implement immersive exhibitions using implicit and explicit proxemic interactions. Using Triggerzones, freely positionable virtual areas in the room, we support the rapid prototyping of interaction concepts and last-minute changes of settings on the exhibition site. We successfully used the tool in two real-world case studies both in the prototyping and in the deployment of two installations, proving its applicability in exhibitions. We also conducted expert interviews showing that Argus Vision fulfills the requirements of the domain of exhibition design. By hiding the application of complex tracking algorithms on the depth data of multiple Kinects behind a user interface, exhibition designers can focus on designing engaging exhibitions using interactive installations, without worrying about complex programming or technological possibilities.

References

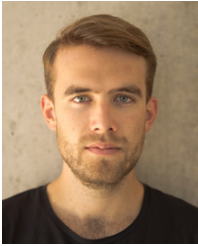
- [1] ART+COM (2008). Kinetic Sculpture. Retrieved January 7, 2019 from <https://artcom.de/en/project/kinetic-sculpture/>.
- [2] ART+COM (2010). Salt Worldwide. Retrieved January 7, 2019 from <https://artcom.de/en/project/salt-worldwide/>.
- [3] ART+COM (2014). MICROPIA. Retrieved January 7, 2019 from <https://artcom.de/en/project/micropia/>.
- [4] Atelier Brückner (2016). Erlebnis Europa – Europa Experience. Retrieved April 03, 2018 from <http://www.atelier-brueckner.de/en/projects/erlebnis-europa-europa-experience>.
- [5] Cafaro, F., Panella, A., Lyons, L., Roberts, J. & Radinsky, J. (2013). I see you there!: developing identity-preserving embodied interaction for museum exhibits. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. New York, NY, USA: ACM. pp. 1911–1920. <https://doi.org/10.1145/2470654.2466252>.
- [6] Electroland (2006). Enteractive. Retrieved January 7, 2019 from <https://www.electroland.net/#/enteractive/>.
- [7] Greenberg, S., Marquardt, N., Ballendat, T., Diaz-Marino, R. & Wang, M. (2011). Proxemic interactions: the new ubicomp?. *ACM Interactions* 18(1), pp. 42–50. <https://doi.org/10.1145/1897239.1897250>.
- [8] Honauer, M. (2013). Designing Device-less Interaction A Tracking Framework for Media Art and Design. In Boll, S., Maaß, S. & Malaka, R. (Hrsg.): *Mensch & Computer 2013 – Workshopband*. München: Oldenbourg Verlag. S. 535–538.
- [9] Hughes, P. (2010). *Exhibition Design*. London, UK: Laurence King.
- [10] Ju, W. & Leifer, L. (2008). The Design of Implicit Interactions: Making Interactive Systems Less Obnoxious. *Design Issues* 24(3), pp. 72–84. <https://doi.org/10.1162/desi.2008.24.3.72>.
- [11] Klein Dytham architecture (2002). Bloomberg ICE. Retrieved January 7, 2019 from <http://www.klein-dytham.com/bloomberg/>.
- [12] Kortbek, K. J. & Grønbaek, K. (2008). Communicating art through interactive technology: new approaches for interaction design in art museums. In *Proceedings of the 5th Nordic conference on Human-computer interaction: building bridges (NordiCHI '08)*. New York, NY, USA: ACM. pp. 229–238. <https://doi.org/10.1145/1463160.1463185>.
- [13] Kossmann, H. & de Jong, M. (2010). *Engaging Spaces: Exhibition Design Explored*. Amsterdam, Netherlands: Frame.
- [14] Kubitz, T., Thullner, S. & Schmidt, A. (2015). VEIL: A Toolkit for Editing Multimedia Content of Interactive Installations On-site. In *Proceedings of the 4th International Symposium on Pervasive Displays (PerDis '15)*, New York, NY, USA: ACM. pp. 249–250. <https://doi.org/10.1145/2757710.2776806>.
- [15] Marquardt, N., Diaz-Marino, R., Boring, S. & Greenberg, S. (2011). The proximity toolkit: prototyping proxemic interactions in ubiquitous computing ecologies. In *Proceedings of the 24th annual ACM symposium on User interface software and technology (UIST '11)*. New York, NY, USA: ACM. pp. 315–326. <https://doi.org/10.1145/2047196.2047238>.
- [16] Maye, L. A., McDermott, F. E., Cioffi, L. & Avram, G. (2014). Interactive exhibitions design: what can we learn from cultural heritage professionals?. In *Proceedings of the 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational (NordiCHI '14)*. New York, NY, USA: ACM. pp. 598–607. <https://doi.org/10.1145/2639189.2639259>.
- [17] MESO (2011). Reactive Light Effect for Pedestrian Tunnel. Retrieved January 7, 2019 from <https://meso.design/en/projects/city-of-aarau-reactive-light-effect-for-pedestrian-tunnel>.
- [18] Monaci, G., Gritti, T., Vignoli, F., Walmlink, W. & Hendriks, M. (2011). Flower Power. In *Proceedings of the 19th ACM International Conference on Multimedia*. New York, NY, USA: ACM. pp. 909–912. <https://doi.org/10.1145/2072298.2071900>.
- [19] Petrelli, D., Cioffi, L., v. Dijk, D., Hornecker, E., Not, E. & Schmidt, A. (2013). Integrating material and digital: a new way for cultural heritage. *interactions* 20(4), pp. 58–63. <https://doi.org/10.1145/2486227.2486239>.
- [20] Schwarz, U. (2001). Entstehungsphasen einer Ausstellung. In Schwarz, U. & Teufel, P. (Hrsg.): *Museografie und Ausstellungsgestaltung*. Ludwigsburg, Germany: avedition. pp. 16–37.
- [21] Seyed, f., Azazi, A., Chan, E., Wang, Y. & Maurer, F. (2015). SoD-Toolkit: A Toolkit for Interactively Prototyping and Developing Multi-Sensor, Multi-Device Environments. In *Proceedings of the 2015 International Conference on Interactive Tabletops & Surfaces (ITS '15)*. New York, NY, USA: ACM. pp. 171–180. <https://doi.org/10.1145/2817721.2817750>.
- [22] Skowronski, M., Wieland, J., Borowski, M., Fink, D., Gröschel, C., Klinkhammer, D., Reiterer, H. (2018). Blended Museum: The Interactive Exhibition “Rebuild Palmyra?”. In *Proceedings of the 17th International Conference on Mobile and Ubiquitous Multimedia (MUM '18)*. New York, NY, USA: ACM, pp. 529–535. <https://doi.org/10.1145/3282894.3289746>.
- [23] TAMSCHICK (2014). TIME MACHINE. Retrieved January 7, 2019 from <http://www.tamschick.com/en/projects/time-machine/>.
- [24] tisch13 (2016). KUKA Brand Experience. Retrieved January 7, 2019 from <https://www.tisch13.com/en/projects/kuka-brand-experience/>.

- [25] Weiser, M. (1991). The Computer for the 21st Century. *Scientific American* 265(3), pp. 94–104.
- [26] Wolf, K., Abdelhady, E., Abdelrahman, Y., Kubitz, T. & Schmidt, A. (2015). meSch: tools for interactive exhibitions. In *Proceedings of the Conference on Electronic Visualisation and the Arts (EVA'15)*. London, UK: BCS. pp. 261–269. <http://dx.doi.org/10.14236/ewic/eva2015.28>.

**Harald Reiterer**

Human-Computer Interaction Group,
University of Konstanz, Konstanz, Germany
harald.reiterer@uni-konstanz.de

Bionotes

**Moritz Skowronski**

Human-Computer Interaction Group,
University of Konstanz, Konstanz, Germany
moritz.skowronski@uni-konstanz.de

Moritz Skowronski (B. Sc.) studies Computer and Information Science at the University of Konstanz and is student assistant at the Human-Computer Interaction Group of Prof. Dr. Harald Reiterer. His main interests are generative design, digital art and the design of interactive exhibitions.

**Daniel Klinkhammer**

Human-Computer Interaction Group,
University of Konstanz, Konstanz, Germany
daniel.klinkhammer@uni-konstanz.de

Daniel Klinkhammer (M. Sc.) is research assistant at the Human-Computer Interaction Group of the University of Konstanz. In his research he focuses on the design and evaluation of interactive exhibitions. In addition, he is involved in various research projects dealing with User Experience Design and the support of creative processes.

Professor Dr. Harald Reiterer is full professor for Human-Computer Interaction at the Computer and Information Science Department of the University of Konstanz. His main research interests include different fields of Human-Computer Interaction, like Interaction Design, Usability Engineering, and Information Visualization. For more information about ongoing projects and actual publications visit the website of his research lab: hci.uni-konstanz.de