Erschienen in: Journal of Visual Languages & Computing. 2018, 46, pp. 35-42
https://doi.org/10.1016/j.jvlc.2017.10.002

ContextuWall: Multi-site collaboration using display walls☆

Matthias Klapperstueck [a,*], Tobias Czauderna [a], Cagatay Goncu [a], Jaroslaw Glowacki [a], Tim Dwyer [a], Falk Schreiber [a,b], Kim Marriott [a]

[a] *Faculty of Information Technology, Monash University, Melbourne, Australia*
[b] *Department of Computer and Information Science, University of Konstanz, Konstanz, Germany*

A B S T R A C T

The emerging field of Immersive Analytics investigates how novel display and interaction technologies can enable people to better explore and analyse data and complex information. Collaboration is a crucial aspect of Immersive Analytics. In this paper we present ContextuWall, a system for interactive local and remote collaboration using touch and mobile devices as well as displays of various sizes. The system enables groups of users located on different sites to share content to a jointly used virtual desktop which is accessible over a secured network. This virtual desktop can be shown on different large displays simultaneously, taking advantage of their high resolution. To enable users to intuitively share, arrange as well as annotate image content, a purpose-built client software has been developed and can easily be adapted with plug-ins for existing data analytics software. We show exemplary use cases and describe the system architecture and its implementation.

1. Introduction

Visual Analytics [1] is an important element in analytical reasoning and decision making and an integral part in the emerging field of Immersive Analytics [2]. Immersive Analytics investigates how to support data analysis and visualisation using immersive display and interaction devices. It has a focus on supporting collaborative data exploration and decision making.

In this paper we focus on collaboration as a way for groups of people to work together on a given problem or dataset to achieve a successful understanding of those using visual analytics [3,4]. Collaboration is increasingly important, in particular in science [5]. Fig. 1 shows the time-space matrix giving an overview of the four common ways to collaborate [6]. Users can be at the same place at the same time utilising local setups such as displays and input devices to work with the data using direct face-to-face communication. Two groups of users can be located at different places and collaborate on the same problem at the same time using video/voice communication as well as a similar setup on the remote site to synchronise the content they are working on. Both scenarios can also happen at different time points where the applied tools need to store their state when a group locally or remotely leaves the discussion only to rejoin at a later time.

In many different areas local and remote visual collaboration is being used already. In medicine doctors share images of medical data such as MRI or X-ray with their colleagues to discuss further treatments. In industry photos of equipment, blueprints and other content might be shared with colleagues at a remote site. In science, data and images are shared with remote collaborating labs to discuss next steps for experiments. In all of these examples collaborative interactive exploration of the shared content is core. Although interactive exploration is usually used in the context of utilising computational power to dynamically create and explore content, it can also be seen as a way for users to share, arrange, and annotate static content to explore and understand the images using interactive methods and to help groups in their decision making.

The client interface is therefore a key part between the user and the system as it mediates such interactive exploration. Many tools exist to do screen sharing with support for annotations but they are either limited in the number of users working on shared content at the same time, or limited in the way they share content with remote sites, or limited to specialised hardware (see Section 3 for a list of investigated tools). Another key aspect is data privacy when dealing with sensitive data that cannot not be shared with online cloud services [7].

☆ This document is based on the 2016 BDVA conference paper *"ContextuWall: Peer collaboration using (large) displays"* https://doi.org/10.1109/BDVA.2016.7787047.

* Corresponding author.
E-mail addresses: matthias.klapperstueck@monash.edu (M. Klapperstueck), tobias.czauderna@monash.edu (T. Czauderna), cagatay.goncu@monash.edu (C. Goncu), jwglo1@student.monash.edu (J. Glowacki), tim.dwyer@monash.edu (T. Dwyer), falk.schreiber@uni-konstanz.de (F. Schreiber), kim.marriott@monash.edu (K. Marriott).

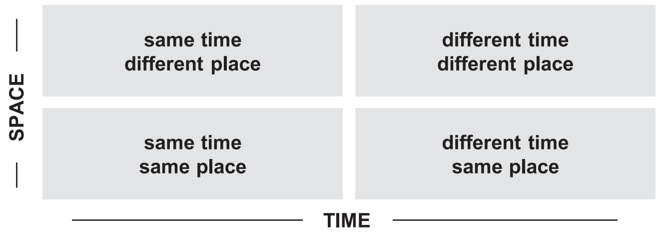

Fig. 1. Time-space matrix for collaboration/group work showing the four possibilities of same time/different time and same place/different place combinations, adapted from Johansen [6].

Here we present ContextuWall, an open source collaboration tool that connects multiple remote sites, runs under full control in the organisation's network, and supports are variety of available visualisation and display technologies ranging from professional and highly priced (e. g. Cave Automatic Virtual Environment - *CAVE* [8]) to consumer grade hardware. In addition, ContextuWall supports all four elements in the time-space matrix (Fig. 1). To facilitate the wide range of available hardware the system is designed as a client-server-display framework with: (1) the client being the interface for the user to share content; (2) the server synchronising its content with multiple remote sites, and (3) the display rendering the content and corresponding annotations. Our main contribution is the synchronised interactive exploration as well as the simultaneous manipulation of displayed content from multiple sites supporting multiple display configurations. The key novelty is the design of the interacting client to intuitively share and annotate the content to facilitate multiple display configurations [9].

Section 2 will give example use cases to further support our motivation, followed by related work regarding existing systems to support collaboration in Section 3. Section 4 describes the implementation in detail which is followed by future work in Section 5 and concluding remarks in Section 6.

2. Results and use cases

The following use cases present applications where the ContextuWall system can show its potential. The focus of the system is on sharing of images with multiple sites, local and remote, as well as annotation of this content to support collaborative work. In this context high resolution content is defined as images with more than 8 megapixels (4K UHD resolution).

2.1. Local and remote collaboration

For success in the biosciences collaboration is essential [10], and this scenario from the life sciences involves a lab meeting between two collaborating groups of biological scientists, located on different sites. Both groups use large display walls to show high resolution content at the same time as shown in Fig. 2.

The displays are connected by two ContextuWall servers each driving one display respectively. Additionally they might use a video conferencing system to enable separate audio/video communication.

One group uses VANTED [11], a JAVA based software for visualising and analysing experimental data in the context of biological networks [12]. They want to share a network with experimental data mapped on it to discuss data, take advantage of complementary expertise within the two groups or plan further experiments based on the data.

An add-on has been implemented to enable VANTED to share images of the entire network or selected parts on large display walls using the ContextuWall system. In comparison to a normal

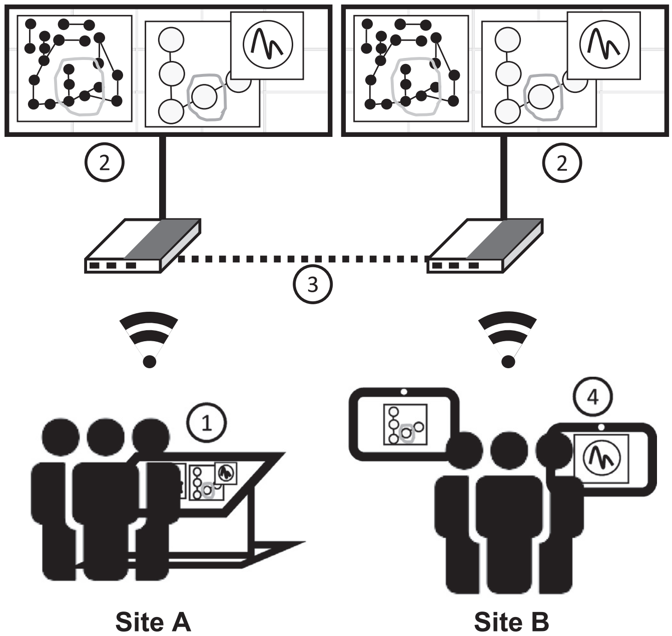

Fig. 2. Application of the ContextuWall system where site A uses a ContextuWall enabled software to share high resolution images (1) on their display wall (2) as well as sending them to the remote site over a secure connection (3). The remote site also has a large display (2) and focuses the discussion on details using annotations created on touch enabled devices (4).

screen sharing session, where the resolution is limited to the device sharing the screen, the VANTED client can always share high resolution images as parts of the network can be selected and rendered as high quality images. This is important as biological experiments often produce large amounts of data which need to be visualised.

A scientist located on *Site A* will share the entire network including related data as a high resolution image. The image can be zoomed and panned to get an overview of the network and its mapped data. The scientists on *Site B* are then interested in a particular region of this network which they would like to investigate in more detail. Using their client they annotate a region of interest. This can be seen by the colleagues on *Site A*, where they select this particular region of the network in the analysis tool VANTED. The selection is then shared as another high resolution image to the groups. The entire network and the selected part of it can be viewed simultaneously on both sites and arranged next to each other creating a factual context. After some discussion within or between the two groups colleagues on *Site B* want, for example, to see more details about a particular element in the network. *Site A* can now create another even more detailed image of this element. All images can be inspected on the displays on both sites at the same time giving the scientists a method to move from an overview to a detailed view of the network without losing the context (see Fig. 3 for a similar workflow).

2.2. Collaboration with on-site environments

This scenario comprises situations when information captured on-site needs to be communicated promptly with groups of specialists located remotely, e. g. at universities, company offices or workshops. It has been shown that the communication efficiency increases by using images and annotations [15].

A company with remote sites such as a mining site, a gas plant, or a production plant might experience a failure in some on-site equipment. Related work proposes live video streaming [16] with annotations to one site whereas our work is focusing on sharing

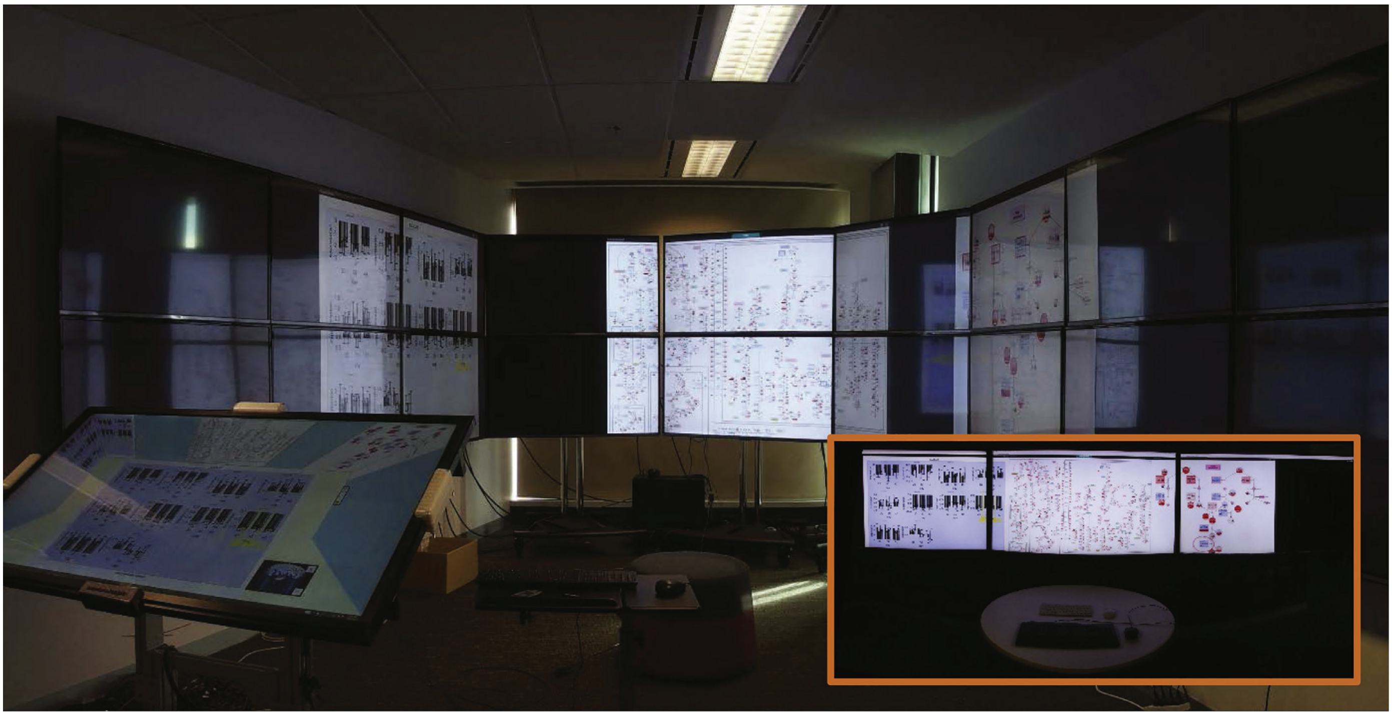

Fig. 3. Use case scenario showing the PerceptuWall, a consumer grade high resolution display with eighteen 4k screens and a remotely connected simple display wall with three 4k screens (lower right) where scientists on one site use the VANTED client to share an image of an entire biological network and more detailed images of selected areas. The remote site uses the ContextuWall client to arrange and annotate the shared images. Both walls show the same content. The images shown are based on information from the MetaCrop database [13,14].

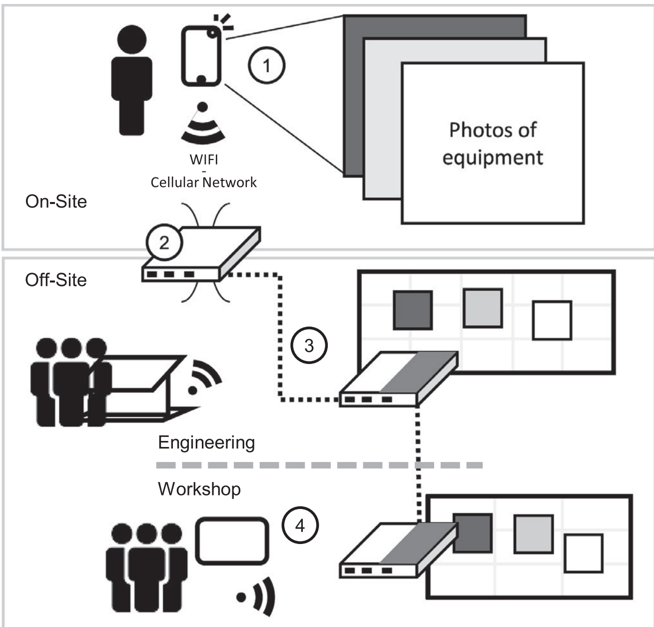

Fig. 4. Application of the Android ContextuWall client where a worker on-site is sharing a series of photos (1) of malfunctioning equipment to the engineering office (3) using an internet connected ContextuWall server (2). The engineers connect to the workshop site (4) for further discussions using annotations to solve the problem.

several images and annotations to multiple sites using large displays.

Fig. 4 shows an application where the worker on-site contacts the engineering office and shares a photo of the malfunctioning part or a photo of a hand drawn sketch on a sheet of paper. Possible additional verbal communication can be conducted using mobile phone or VoIP software. After a brief discussion the engineering office decides to contact the workshop to further discuss the problem. The workshop located at another site can join the discussion by opening a connection to the existing ContextuWall session to see the same content on their display as engineering. While the engineering office and the workshop have large displays to arrange the images including their annotations the mobile client is limited to displaying one full screen image at a time due to its display size. To circumvent this limitation an additional list of shared images enables the user to select which image to view. The on-site worker can leave the session at any time and rejoin later for further discussion or to view the results or instructions.

In another scenario archaeologists at an excavation site or biologists on a field trip make an important discovery and need to discuss details about their discovery instantly with experts around the world. Being able to share images and to annotate these images provides means for planning the further advancement on the excavation site or during the field trip.

3. Related work

Research on local collaboration/group work incorporating large shared displays and according interaction methods has been conducted since the 1980s. Examples for different approaches are, e.g. Colab [17], i–LAND [18], Augmented Surfaces [19], iRoom [20], MultiSpace [21], Table-Centric Interactive Spaces [22], and IMPROMPTU [23]. Brown et al. [24] give an overview of more recent developments.

Collaborative Annotations on Visualizations (*CAV*) [25] employs a concept similar to ContextuWall. A client application for tablet PCs supports sharing of images and image annotations; clients are connected to a visualisation server. However, CAV focuses on same time-different place and different time-different place collaboration. In comparison to ContextuWall, CAV does not show support of large displays.

In recent years numerous software and/or hardware products have been made available supporting local and remote collaboration. In the following application areas are listed including examples of products either being cloud based [7], requiring specialised hardware, or having other limitations (see below):

- online text document sharing using, e. g. Evernote,[1] Google Drive[2]
- online image sharing + image annotation using, e. g. A.nnotate,[3] Marqueed,[4] RealtimeBoard[5]
- screen sharing using video conferencing software such as Google Hangouts,[6] Skype,[7] TeamViewer,[8] Zoom[9]
- source code sharing using, e. g. Git repositories, Subversion repositories
- interactive whiteboards such as Hitachi Interactive Whiteboard,[10] Smoothboard Air[11]
- meeting room collaborative environments such as Cisco Spark Board,[12] DisplayNote,[13] Epson MeetingMate,[14] Google Jamboard,[15] Intel Unite,[16] Microsoft Surface Hub[17]
- large display environments for collaboration such as Jupiter Canvas,[18] Mersive Solstice,[19] SAGE2 [26]

Cloud based text document, image or source code sharing solutions often lack privacy. Mostly a sign-in is required, however, data is not necessarily private unless you pay for the service but the data is still stored on non-private servers. This might be an issue in certain collaborations when data cannot be disclosed. For screen sharing solutions the resolution depends on the screen resolution of the source device sharing its screen. In addition only one person at a time can manipulate the screen. Interactive whiteboards require a specialised hardware setup and do not necessarily support remote collaboration. Commercial products such as Intel Unite, DisplayNote, Jupiter Canvas, and Mersive Solstice are proprietary and to the best of our knowledge offer limited possibilities for extension or integration of own applications. SAGE2 can share content with multiple sites but sharing has to be done explicitly for each remote site and every shared media (applications, PDF documents, images, videos). Furthermore participating sites need to have the same display configuration and resolution for a synchronised layout of content shown, and annotation of content is not possible.

4. Methods and implementation

In this section we present the architecture of ContextuWall which includes a description of the overall idea, a walk through of a typical workflow, and describe each component involved in the system.

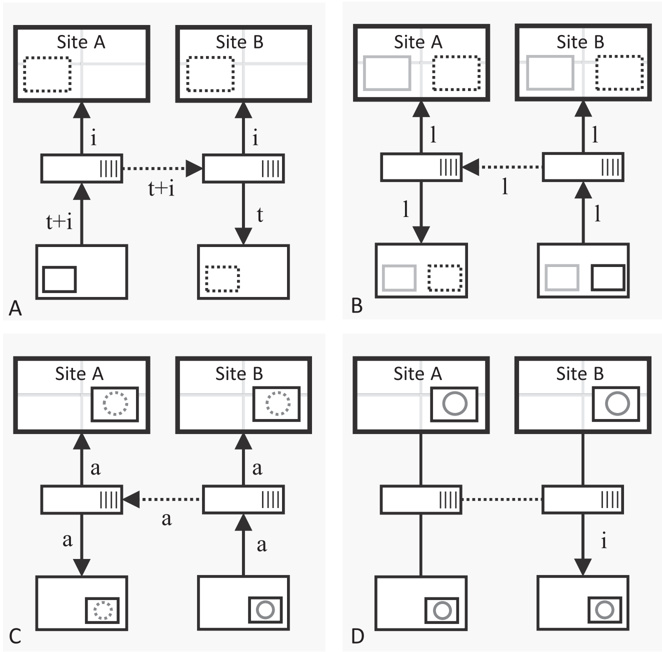

Fig. 5. Typical workflow using the ContextuWall client-server-display system: A) upload/sharing of content, the thumbnail and the image are distributed to all displays, servers, and clients; B) interaction with content, the new location and size of the image is distributed to all displays, servers, and clients; C) annotation is added to the image which is shared with the servers and shown on all clients and displays; D) annotated images can be downloaded throughout a session. Arrows from a client to a server indicate a user action, an arrow from a server to a client indicates an update to the client. *(Legend: i=image, t=thumbnail, a=annotation, l=location).*

4.1. Collaborative idea

We wanted to create a system that supports all four types of collaboration depicted in the time-space matrix shown in Fig. 1. It should support local collaboration between groups of users but also enable remotely located collaborators to join the discussion and get access to shared content. Every site should be able to leave and rejoin the discussion at any time as well as continue the discussions separately.

A way to support all four types of collaboration is to use static content such as images as well as having the ability to annotate these images to highlight points of interest to support visual analytics and reasoning. The content is shown on a jointly used virtual desktop presented to all participants and can be manipulated by each participating site simultaneously. Using large displays enables users to show several high resolution images and to arrange these images without overlap.

The system should also support various kinds of available display hardware from professional (e.g. CAVE2[20]) to cheaper consumer grade products such as simple multi-screen solutions.

4.2. Workflow

Fig. 5 shows a typical workflow using the ContextuWall client-server-display system. Each of the four steps show two remote sites apart from each other linked by a server connection. All communication, shown as lines, is encrypted. Let the left side be *Site A* and the right side be *Site B*. *(A)* shows the start of a new session. The display is empty and a user at *Site A* shares an image using the client. The client creates a thumbnail of that image and sends the image and the thumbnail to the server. The server sends

[1] https://evernote.com/.
[2] https://www.google.com/drive/.
[3] http://a.nnotate.com/.
[4] http://www.marqueed.com/.
[5] https://realtimeboard.com/.
[6] https://hangouts.google.com/.
[7] https://www.skype.com/.
[8] https://www.teamviewer.com.
[9] https://zoom.us/.
[10] http://www.hitachi.com.au/dps/products/starboards.html.
[11] http://www.smoothboard.net/.
[12] https://www.cisco.com/c/en/us/products/collaboration-endpoints/spark-board/.
[13] https://www.displaynote.com/.
[14] http://www.epson.com.au/products/Projector_interactive_meetingmate/.
[15] https://gsuite.google.com/jamboard/.
[16] http://www.intel.com/unite.
[17] https://www.microsoft.com/microsoft-surface-hub/.
[18] http://www.jupiter.com/solutions/Canvas/.
[19] https://www.mersive.com/products/solstice/.
[20] http://www.mechdyne.com/hardware.aspx?name=CAVE2.

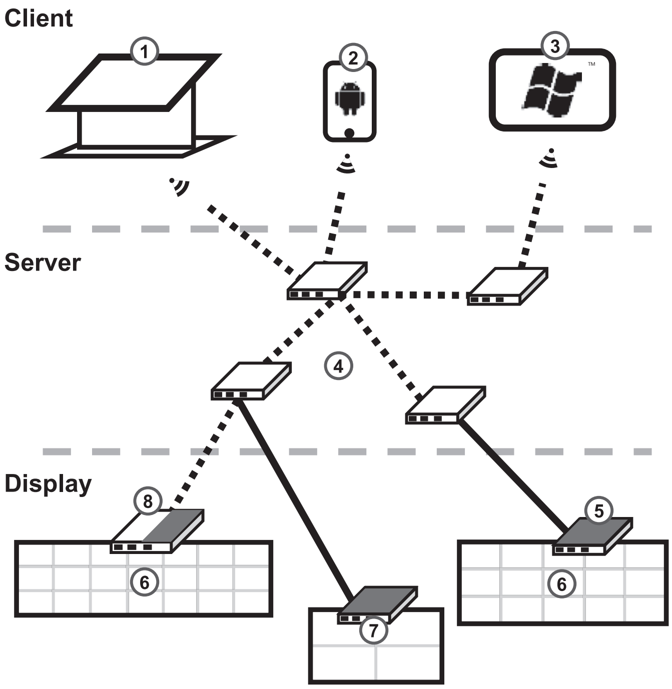

Fig. 6. Overall architecture of ContextuWall comprising of different clients (top) such as touch tables (1), smartphones (2) or tablets/notebooks (3); servers (middle) (4); various display architectures (bottom) including cluster displays ((5) head node, (6) cluster display), standard PC setups (7) or mixed systems (mixed head node (8) with the ContextuWall server and the display software driving a cluster display (6)).

the image to its connected display and sends the image and the thumbnail to the remote server. The remote server will send the image to its connected display as well as the thumbnail to its connected client. The user at *Site B* decides to move the image *(B)* by using the touch interface of the client. During movement of images updates of the position are sent in real time to the servers, clients, and displays on both sites. During a discussion *Site B* adds an annotation to the image *(C)* to highlight a certain issue or to focus the discussion to a certain area in the image. While the annotation is drawn a continuous update of the annotation image is sent to the servers and can be seen on the remote client as well as on the displays on both sites. At any point during the meeting *(D)* both sites can download the shared image including its annotation to a local storage to keep it for later discussion. Although this workflow shows only one site sharing the image and the other site moving and annotating the image, it is in fact possible that both sites can share, move, and annotate images displayed on the screens simultaneously, unlike a screen sharing session where only one site has control about the shared content.

4.3. System architecture overview

To create a distributed system as well as having certain independence of the participating hardware setup on each site we decided to create a three-tier architecture comprising a) clients, b) servers, and c) displays.

Fig. 6 gives an overview of all components involved. At the top a variety of clients can be seen. They are used to share images, to control the arrangement and size of images, as well as to graphically annotate the content. Clients can be any kind of smartphone (2), tablet, notebook (3) or touch table computer (1), running at least Microsoft Windows® 8 or Android 5.0 Lollipop. These devices connect to one of the servers via LAN, WLAN or a cellular network.

In the middle the ContextuWall server network is shown (4). Multiple servers are connected in a tree structure. The data and instructions comprising images, annotations, movement and zooming of images are synchronised within the server network. Servers can be dynamically added to the network synchronising their current state automatically with the server network on start-up or if the servers rejoin the network. The number of servers connected to the network is only limited by the server performance and the network connection speed.

At the bottom different types of displays currently supported by ContextuWall are shown. Displays can be added and removed dynamically. Multiple configurations are possible and are described briefly. (6) represents a cluster display such as the CAVE2 where multiple computers drive groups of screens to create a large display. Usually cluster displays have a head node (5) that controls and synchronises the content. The ContextuWall display software is executed on the head node which is connected to a ContextuWall server to receive the data it has to render. (7) represents a display with multiple screens attached to a standard PC running the display software being connected to a ContextuWall server. The last display type shows a variant where the ContextuWall server as well as the display software runs on the same computer (8) driving a cluster display (6).

Secured WebSockets are used for communication between server, display, and client using JSON.[21] In addition, sessions are also password protected to restrict access to sensitive information. The dotted lines show client-server and server-server connection whereas the solid line shows connections between display software and the ContextuWall server. In case of (8) the connection is using a local unencrypted network socket.

The communication protocol between server-server and server-client has been kept simple and lightweight and has been optimised for large image transfer and state synchronisation (image position, image size, annotations). The whole system does not depend on very high speed bandwidth such as GBit networks as the only high volumetric data is the uploaded images. We use asynchronous streaming between servers to keep the system responsive as well as to keep a copy of each image on each server as backup and synchronisation source for newly connected servers. Images are transferred as such without transcoding to save processing time. In case a server is connected using low bandwidth such as WiFi or 3G images are always transferred as thumbnails before the actual file stream begins.

Annotations are lossless compressed images that are transferred as a whole each time the annotation is updated and represent a pixel accurate overlay over the original image. To exchange annotations we use the same file streaming technique as for transferring the original images. The size of the annotations is significantly smaller since they are mostly used to highlight certain areas in the original image.

All other commands such as moving an image, scaling an image, deleting annotation layers, only contain reference IDs and coordinates. These packages can be transmitted utilising a high update rate between the whole server-client network. The actual update rate on the display is only limited by the rendering speed of the computer driving the display.

4.4. Client

The ContextuWall client can be a touch enabled, portable device to support an intuitive workflow staying mobile throughout the session. Using touch tables with high resolution screens enables users to create detailed annotations of images.

A C# implementation was created for devices running Microsoft Windows® 8. Encrypted communication between client and server

[21] http://www.json.org/.

Fig. 7. Two modes of the Windows ContextuWall client can be seen at the top whereas the graphics at the bottom show the application of each mode. The left shows the flat mode used for standard displays and display walls whereas the right shows the immersive mode primarily used for surrounding displays such as the CAVE. The upper screen area shows the interaction area to move, zoom and arrange images on the display. An image can be dragged to the annotation area below the interaction area where graphical annotations can be added. The ContextuWall widget shown in the lower right of each image can be used to drag and drop images from the file manager.

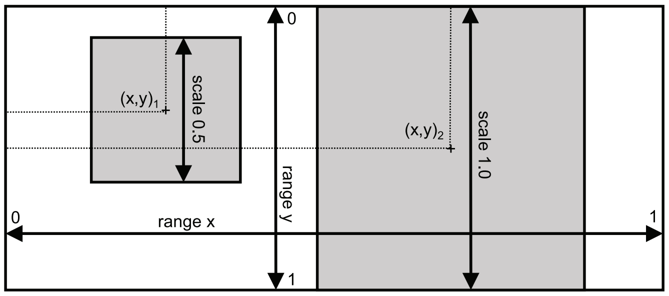

Fig. 8. Using normalised coordinates within the range of [0..1] and a relative scale factor within the range of]0..*n*] to establish an independence from all displays connected to the ContextuWall system. The position of images is related to the centre of the images. A scale factor of 1 renders the image to the full height on each display.

is performed via JSON using the socket.io[22] library. Clients have to identify themselves by providing password credentials upon connection.

The client can be configured for two modes as shown in the two screenshots in Fig. 7. The top left screenshot shows the flat mode where the user is presented with a screen overlay on the upper side of the desktop where all the shared images are shown as thumbnails and appear with respect to their position on the display. This mode is used for standard displays and display walls with high resolutions. The user can use swipe gestures to share a screenshot of the current Windows desktop content with the display. Depending on the end position of the gesture the image is positioned on the display respectively. If the client is not touch enabled a mouse-drag gesture can be used instead.

The top right image in Fig. 7 shows the immersive mode used for CAVE display systems where the client is located in the centre of the display (see also Fig. 3). In this mode the user is presented with three bars on the left, top, and right side of the screen. The user's swipe action direction to share an image will correspond with the position of the image on the surrounding display.

In addition to taking screenshots the client provides a ContextuWall widget to support drag and drop gestures to share prepared images from local storage such as photos or images created by applications beforehand. First the client sends a small thumbnail to the servers which can be rendered very quickly on the connected displays followed by the full size image. Using this two-step method provides a better user experience with immediate feedback.

After the user has shared the image it also appears on all connected clients. Images can be moved using single-finger panning gesture to place it on the desired position and resized using two-finger pinch gesture. In case of image overlap users can change image ordering by bringing images to the front or putting them to the back.

Users can then drag an image to the annotation area of the client where graphical enrichments with different colours and drawing styles can be added to that image using finger, stylus or mouse (see Fig. 7). The annotations immediately appear on the displays and on the other clients. A user can download the full size image at any time including its annotations. An annotation created by a user on a client is bound to a particular image and can be removed by that same client again as annotations are linked to that client with a client ID. Furthermore, any client can remove images or all annotations linked to an image which will then disappear from all displays and clients.

An implementation of the client protocol has been created for the Android platform that enables users to share photos taken with the camera to local and remote sites. Users can use ContextuWall clients to annotate and discuss these images. The Android client supports all ContextuWall features including creating annotations and rearranging images on the displays.

To show the feasibility to link third party software with ContextuWall an add-on for VANTED has been developed that implements parts of the client API. This client is limited in its functionality as it only supports sharing of images. A use case showing an application of this add-on is presented in Section 2.

4.5. Server

To achieve platform independence the server is implemented in JavaScript and executed in node.js. Its main purpose is to communicate with the clients, the displays as well as keep the current state synchronised between connected servers.

The server stores the current state of the ContextuWall session context in a data model. It contains metadata for each image, its thumbnail, its current position and scale factor, as well as all annotations drawn on that image. The actual image which is sent by clients is stored on the server hard drive and not kept in memory to minimise the memory footprint. Only the file reference is stored in the data model. Each annotation sent by a client is also stored as a separate image file on the server hard drive and is linked to its source client by an additional ID.

To create independence from local and remote display configurations and resolutions the position and size of each image is stored as normalised coordinates using a [0..1] range for x/y position and a]0..*n*] range for scale factors where 1.0 corresponds to the height of the display as shown in Fig. 8. The height was chosen as the scale factor since the majority of display walls have an aspect ratio greater than one.

4.6. Display

Throughout project development multiple implementation of the display software was performed for different variants of available hardware including smaller displays using standard PC / multiple screen systems up to large displays using cluster display technology.

[22] http://socket.io/.

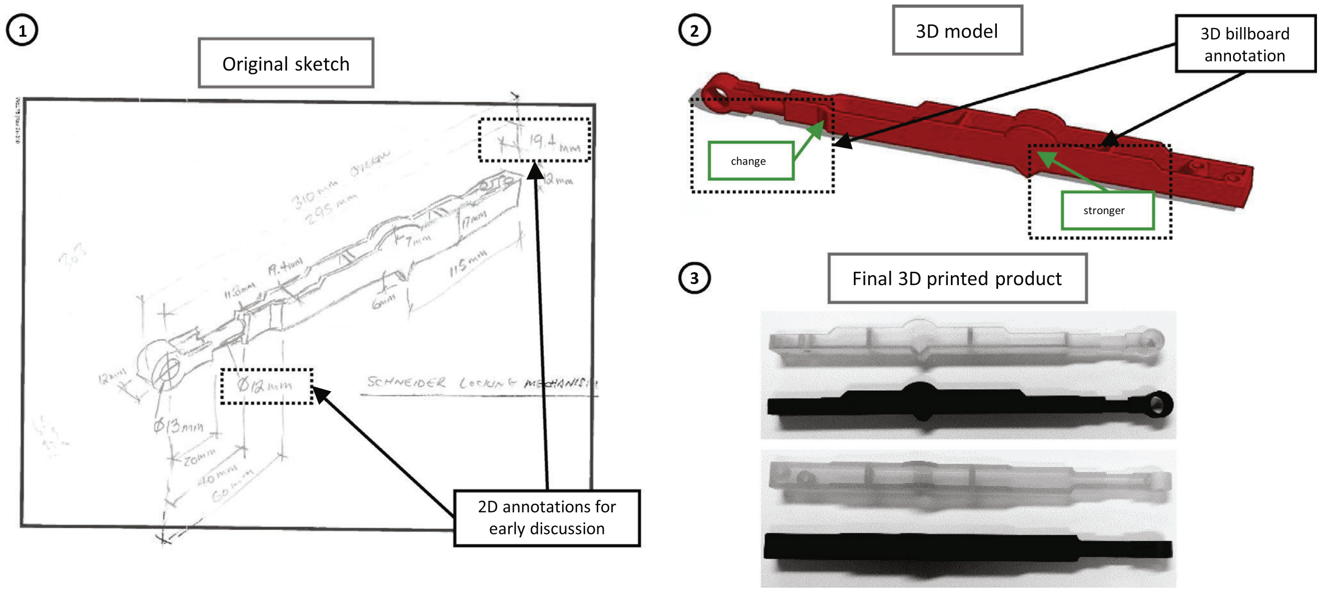

Fig. 9. This figure shows the extension of the second use-case as discussed in Section 2 regarding collaboration with on-site environments (see Fig. 4). Until now only discussions using static 2D content can be performed, as shown in (1). Multiple sites can simultaneously collaborate on early discussions about equipment before the 3D modelling process starts. Future work will investigate ways to integrate the ContextuWall system to support the entire pipeline: (1) early discussion stage, (2) advanced discussion stage which includes 3D modelling, (3) final product. This includes multi-site discussions on dynamic 3D models by placing billboard annotations inside the 3D model as shown in (2). Therefore the applicability of different input devices and methods, from touch table to 3D wand devices, needs to be investigated for intuitive remote collaboration on 2D and 3D content.

The display renders the images and annotations received from a connected server as well as updates location and scaling of displayed images. Images and annotations are rendered in layers where each annotation layer corresponds to the client that created this annotation.

Although the framework supports independent display resolutions for each participating site in a multi-site collaborative setup the aspect ratio and resolution should be similar to achieve optimal user experience. The display with the smaller aspect ratio will produce more overlap of images and the display with smaller resolution will create less readable content considering the same scaling of the images.

4.6.1. Cluster display

The first implementation involved a cluster display enabling us to benefit from a very high resolution. One of the largest systems available at the time of development was the CAVE2[23] at Monash University. It is a system comprising 80 screens configured in a 20 columns by 4 rows matrix resulting in a total resolution of about 80 megapixels. We developed the display software using the omegalib framework [27] which manages the distribution and synchronisation of the content throughout the display cluster.

4.6.2. Standard PC with multiple screens

The second implementation was created to utilise a standard PC and screen components with high resolution. The display software was developed using the Unity game engine[24] which renders the images and annotations as high resolution textures. In comparison to the CAVE2 cluster display system this setup only uses consumer grade products. A setup comprising a single PC with an ATI W9100 high performance graphics card connected to six 55″ 4k TV screens mounted as a 3 columns by 2 rows matrix with a combined resolution of about 50 megapixels was used with the ContextuWall system for remote collaborations with the CAVE2.

4.6.3. SAGE2 display

Maintaining and improving multiple code bases is time consuming. We were looking for a way to support a multitude of display configurations but only using a minimal set of different software packages to maintain. We therefore aimed to use the already existing SAGE2[25] [26] in-house installations to unify the systems architecture. Its architecture supports arbitrary display configurations starting from single screen up to cluster display environments such as the CAVE2. SAGE2 itself supports different types of remote site collaboration: (1) display mirror: remote sites connect to the same URL to see the content but with the limitation to the same display dimension, and (2) application share: each application or image is shared with each remote site explicitly but with the lack of synchronous layout and the need to repeat the process for each shared item. To overcome these limitations a connection between the ContextuWall server and the SAGE2 server that extends the SAGE2 environment with the features of the ContextuWall system was implemented.

Our different implementations show that it is easy to adapt the system to different hardware environments.

5. Future work

Future work on the ContextuWall system will include improvements for annotations of 2D content as well as support for interactive 3D content.

With regard to 2D content the ContextuWall system will include different extensions for an enhanced user experience such as additional annotation features, e. g. towards text annotations; extension of the media which can be shared and annotated, e. g. towards PDF files and videos; or integration of interaction devices such as wands and Gyration Air Mouse[26] or touch input for direct annotations on large displays.

Until now annotations are restricted to 2D content which e. g. covers use-cases shown in Section 2. Challenges arise when work-

[23] http://www.monash.edu/mivp.
[24] https://unity3d.com/.
[25] http://www.sagecommons.org.
[26] http://www.gyration.com/.

ing with 3D data sets, such as volumetric data and architectural or construction data. A simple 2D annotation layer on top of a fixed 3D dataset is not sufficient as further exploration of the 3D model would lose synchronisation with the above layer, thus hindering ongoing interactive discussion. The ContextuWall system will be extended towards supporting static and dynamic 3D content using standard as well as stereoscopic display technologies to provide an incorporation of mixed 2D and projected/stereoscopic 3D content. To add to the existing overlaying of annotations on top of static shared content we will investigate dynamic annotations as well as collaborative user interaction to support creating and extending annotations in 3D space. We will especially investigate two interaction modes: 1) placement of drawn 2D annotations as textures in 3D space, and 2) integration of brushing annotations in 3D space. The first mode would only require the extension of the interaction client to enable annotation of 3D content by providing a 2D annotation canvas which can then be placed as texture in 3D space. The second mode will make use of 3D tracked wand hardware to enable brushing annotation of the content displayed. Fig. 9 gives an overview of the 2D/3D integration into the collaborative system.

6. Conclusion

We have presented the client-server-display system ContextuWall for collaborative work using large displays. In addition we have demonstrated its applicability showing exemplary use case scenarios.

The ContextuWall system supports all different types of collaborations shown in the time-space matrix of Fig. 1. Users can collaborate locally using touch enabled devices and large displays to share and annotate high resolution images to discuss and solve problems. Remote collaborators can join these discussion sessions at any time to continue work on already shared images or share and annotate new images as well. Besides being able to work together at the same time it is also possible to continue work after one site has left a session.

ContextuWall extends the classical two site collaboration to a multiple site collaboration enabling users to work together simultaneously on a jointly used virtual desktop. A three-tier infrastructure provides flexibility to incorporate a variety of different technologies including consumer hardware, e.g. being able to use mobile devices to share and annotate photos taken on-site decreases time for problem identification and problem solving.

To enhance the user experience a touch-enabled client was implemented to support intuitive work with large display systems when sharing, arranging and annotating image content.

The software framework behind ContextuWall is open source under MIT license[27] and is freely available from http://www.immersiveanalytics.org/contextuwall.

References

[1] D.A. Keim, F. Mansmann, J. Schneidewind, J. Thomas, H. Ziegler, Visual analytics: scope and challenges, in: S.J. Simoff, M.H. Boehlen, A. Mazeika (Eds.), Visual Data Mining, Lecture notes in computer science, 4404, Springer Berlin Heidelberg, 2008, pp. 76–90, doi:10.1007/978-3-540-71080-6.

[2] T. Chandler, M. Cordeil, T. Czauderna, T. Dwyer, J. Glowacki, C. Goncu, M. Klapperstueck, K. Klein, K. Marriott, F. Schreiber, E. Wilson, Immersive analytics, in: Big Data Visual Analytics (BDVA), 2015, IEEE eXpress Conference Publishing, 2015, pp. 73–80, doi:10.1109/BDVA.2015.7314296.

[3] J. Heer, M. Agrawala, Design considerations for collaborative visual analytics, Inf Vis 7 (1) (2008) 49–62, doi:10.1057/palgrave.ivs.9500167.

[4] P. Isenberg, N. Elmqvist, J. Scholtz, D. Cernea, K.-L. Ma, H. Hagen, Collaborative visualization: definition, challenges, and research agenda, Inf Vis 10 (4) (2011) 310–326, doi:10.1177/1473871611412817.

[5] D. Sonnenwald, Scientific collaboration: a synthesis of challenges and strategies, ARIST 41 (2007), doi:10.1002/aris.2007.1440410121.

[6] R. Johansen, Groupware: Computer Support for Business Teams, Series in communication technology and society, Free Press, New York, NY, USA, 1988.

[7] B. Kahin, J.H. Keller, The self-governing internet: coordination by design, in: B. Kahin, J.H. Keller (Eds.), Coordinating the Internet, MIT Press, 1997, pp. 3–38.

[8] A. Febretti, A. Nishimoto, T. Thigpen, J. Talandis, L. Long, J.D. Pirtle, T. Peterka, A. Verlo, M. Brown, D. Plepys, D. Sandin, L. Renambot, A. Johnson, J. Leigh, CAVE2: a hybrid reality environment for immersive simulation and information analysis, in: Proceedings IS&T / SPIE Electronic Imaging, 8649, 2013, pp. 864903.1–12, doi:10.1117/12.2005484.

[9] J.C. Roberts, P.D. Ritsos, S.K. Badam, D. Brodbeck, J. Kennedy, N. Elmqvist, Visualization beyond the desktop – the next big thing, IEEE Comput. Graph Appl. 34 (6) (2014) 26–34, doi:10.1109/MCG.2014.82.

[10] Editorial, Catalyzing collaboration, Nat. Chem. Biol. 3 (5) (2007) 239, doi:10.1038/nchembio0507-239.

[11] H. Rohn, A. Junker, A. Hartmann, E. Grafahrend-Belau, H. Treutler, M. Klapperstueck, T. Czauderna, C. Klukas, F. Schreiber, VANTED V2: a framework for systems biology applications, BMC Syst. Biol. 6 (1) (2012) 139.1–13, doi:10.1186/1752-0509-6-139.

[12] A. Junker, H. Rohn, T. Czauderna, C. Klukas, A. Hartmann, F. Schreiber, Creating interactive, web-based and data-enriched maps using the systems biology graphical notation, Nat. Protoc. 7 (2012) 579–593, doi:10.1038/nprot.2012.002.

[13] E. Grafahrend-Belau, S. Weise, D. Koschützki, U. Scholz, B.H. Junker, F. Schreiber, Metacrop - a detailed database of crop plant metabolism, Nucleic Acids Res. 36 (1) (2008) D954–D958, doi:10.1093/nar/gkm835.

[14] F. Schreiber, C. Colmsee, T. Czauderna, E. Grafahrend-Belau, A. Hartmann, A. Junker, B.H. Junker, M. Klapperstueck, U. Scholz, S. Weise, Metacrop 2.0: managing and exploring information about crop plant metabolism, Nucleic Acids Res. 40 (1) (2012) D1173–D1177, doi:10.1093/nar/gkr1004.

[15] K. Pritchard, G. Symon, Picture perfect? Exploring the use of smartphone photography in a distributed work practice, Manage. Learn. 45(5) (2013) 561–576, doi:10.1177/1350507613486424.

[16] V. Domova, E. Vartiainen, M. Englund, Designing a remote video collaboration system for industrial settings, in: Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces, in: ITS '14, ACM, New York, NY, USA, 2014, pp. 229–238, doi:10.1145/2669485.2669517.

[17] M. Stefik, G. Foster, D.G. Bobrow, K. Kahn, S. Lanning, L. Suchman, Beyond the chalkboard: computer support for collaboration and problem solving in meetings, Commun. ACM 30 (1) (1987) 32–47, doi:10.1145/7885.7887.

[18] N.A. Streitz, J. Geissler, T. Holmer, S. Konomi, C. Mueller-Tomfelde, W. Reischl, P. Rexroth, P. Seitz, R. Steinmetz, i-LAND: an Interactive landscape for creativity and innovation, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, in: CHI '99, ACM, New York, NY, USA, 1999, pp. 120–127, doi:10.1145/302979.303010.

[19] J. Rekimoto, M. Saitoh, Augmented surfaces: a spatially continuous work space for hybrid computing environments, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '99, ACM, New York, NY, USA, 1999, pp. 378–385, doi:10.1145/302979.303113.

[20] B. Johanson, A. Fox, T. Winograd, The interactive workspaces project: experiences with ubiquitous computing rooms, IEEE Pervasive Comput. 1 (2) (2002) 67–74, doi:10.1109/MPRV.2002.1012339.

[21] K. Everitt, C. Shen, K. Ryall, C. Forlines, MultiSpace: enabling electronic document micro-mobility in table-centric, multi-device environments, in: Proceedings of the First IEEE International Workshop on Horizontal Interactive Human-Computer Systems, TABLETOP '06, IEEE Computer Society, Washington, DC, USA, 2006, pp. 27–34, doi:10.1109/TABLETOP.2006.23.

[22] D. Wigdor, C. Shen, C. Forlines, R. Balakrishnan, Table-centric interactive spaces for real-time collaboration, in: Proceedings of the Working Conference on Advanced Visual Interfaces, AVI '06, ACM, New York, NY, USA, 2006, pp. 103–107, doi:10.1145/1133265.1133286.

[23] J.T. Biehl, W.T. Baker, B.P. Bailey, D.S. Tan, K.M. Inkpen, M. Czerwinski, IMPROMPTU: a new interaction framework for supporting collaboration in multiple display environments and its field evaluation for co-located software development, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '08, ACM, New York, NY, USA, 2008, pp. 939–948, doi:10.1145/1357054.1357200.

[24] J. Brown, J. Wilson, S. Gossage, C. Hack, R. Biddle, Surface Computing and Collaborative Analysis Work, Synthesis lectures on human-centered informatics, Morgan & Claypool Publishers, 2013, doi:10.2200/S00492ED1V01Y201303HCI019.

[25] S.E. Ellis, D.P. Groth, A collaborative annotation system for data visualization, in: Proceedings of the Working Conference on Advanced Visual Interfaces, AVI '04, ACM, New York, NY, USA, 2004, pp. 411–414, doi:10.1145/989863.989938.

[26] L. Renambot, T. Marrinan, J. Aurisano, A. Nishimoto, V. Mateevitsi, K. Bharadwaj, L. Long, A. Johnson, M. Brown, J. Leigh, SAGE2: a collaboration portal for scalable resolution displays, Future Gen. Comput. Syst. 54 (2016) 296–305, doi:10.1016/j.future.2015.05.014.

[27] A. Febretti, A. Nishimoto, V. Mateevitsi, L. Renambot, A. Johnson, J. Leigh, Omegalib: a multi-view application framework for hybrid reality display environments, IEEE Virtual Real. (2014) 9–14, doi:10.1109/VR.2014.6802043.

[27] https://opensource.org/licenses/MIT.