FreeTop: Finding Free Spots for Projective Augmentation

Jan Riemann
Telecooperation Group
TU Darmstadt
D-64289 Darmstadt
riemann@tk.tu-darmstadt.de

Sebastian Döweling
Strategic Projects
SAP SE
D-64283 Darmstadt
sebastian.doeweling@sap.com

Mohammadreza Khalilbeigi
Telecooperation Group
TU Darmstadt
D-64289 Darmstadt
khalilbeigi@tk.tu-darmstadt.de

Florian Müller
Telecooperation Group
TU Darmstadt
D-64289 Darmstadt
mueller@tk.tu-darmstadt.de

Martin Schmitz
Telecooperation Group
TU Darmstadt
D-64289 Darmstadt
schmitz@tk.tu-darmstadt.de

Max Mühlhäuser
Telecooperation Group
TU Darmstadt
D-64289 Darmstadt
max@tk.tu-darmstadt.de

CHI'16 Extended Abstracts, May 07-12, 2016, San Jose, CA, USA
ACM 978-1-4503-4082-3/16/05.
http://dx.doi.org/10.1145/2851581.2892321

Abstract
Augmenting the physical world using projection technologies or head-worn displays becomes increasingly popular in research and commercial applications. However, a common problem is interference between the physical surface's texture and the projection. In this paper, we present FreeTop, a combined approach to finding areas suitable for projection, which considers multiple aspects influencing projection quality, like visual texture and physical surface structure. FreeTop can be used in stationary and mobile settings for locating free areas in arbitrary physical settings suitable for projective augmentation and touch interaction.

Author Keywords
Interactive displays; hybrid physical-digital interaction; peripheral displays; projection; multitouch.

ACM Classification Keywords
H.5.m. [Information Interfaces and Presentation (e.g. HCI)]: Miscellaneous

Introduction
With the advent of inexpensive projection technology, interactive projective augmentation of the physical world has become a common technique in the HCI community [9, 15, 25]. Typically, projection is used to digitally

Figure 1: Colorized projectability map (dark blue=good projectability, red=bad projectability) of a conventional office desk (top) and a mobile paperwork scenario (bottom)

augment surfaces and everyday objects, for instance displaying additional information [15], or self-contained digital content [2, 10]. While practical and widely used to bridge the digital and the physical world, one prominent issue of projection on objects is the interference between the texture (physical and visual) of the physical surface and the projected digital content. Therefore, it is desirable to automatically find areas in the physical world which have properties that facilitate projection and interaction (e.g. light, flat surfaces).

One way to mitigate this issue is to use predefined models, denoting spots for projection. This obviously requires all objects to be static and known in advance. A more dynamic, practically used, approach is to use corner-detection [15] to find areas covered by content (e.g. text). Using corner detection to assess projection quality works well with text, but other content, e.g. pictures, might not have corners to detect. One solution is to use structured-light [2] to find areas suitable for projection. It circumvents the limitations of corner detection but requires significant technical effort. Both approaches suffer from the fact that they do not take the 3D surface of objects into account. This might lead to issues with the legibility and interactability of the projection: For instance, a white box on a white table might look as smooth surface from a top mounted camera, but projection over the edge makes it quite difficult to recognize the content and interact with it. Also, extremely rough surfaces might be unsuitable for touch interaction, but look smooth on the camera image.

In this paper, we argue that employing depth information allows to gain additional insight on the objects' surface and a better understanding of the overall projection surface. Using commodity RGBD cameras (e.g. Kinect), we contribute FreeTop, an approach combining color- and depth-based (surface lightness, visual and physical edges) measures to assign a *projectability score* to every pixel within the camera image. The generated projectability map of a given scene or object can then be used to inform the layout and placement of projection. It can be employed in stationary or mobile (e.g. nomadic pico projectors [10, 12] or AR-glasses) settings for on-surface augmentation. We primarily focus on stationary office settings without limiting FreeTop to this area (see figure 1 for an example of mobile use).

Related Work

Augmented Desktops

Effort has been made to augment paperwork with digital projection or input, thereby allowing the user to interactively add digital content (e.g. annotations, animations, . . .) onto physical documents [9, 15, 16, 25] or next to physical documents [14]. All these systems mostly aim at augmenting paper documents with additional digital facilities and do not focus on self-contained digital objects (e.g. a fully digital document). Despite taking – to some extent– possible interference between physical and digital content into account (e.g., FACT [15] uses corner detection to identify areas covered with text), most of the digital content targeted in these works is related to a physical object it is projected on (e.g. annotations, . . .), so a certain amount of interference is tolerable. To reduce interference, manual user-defined projection areas have been considered [26]. This, however, requires active user intervention and does not automatically account for changes in the surface.

For general-purpose tabletop-systems system were proposed to to find an uncluttered area suitable for displaying content [21]. The Display Bubbles [2] system uses bubble shaped display areas to fill the areas between physical obstacles and thereby avoid interference. Both

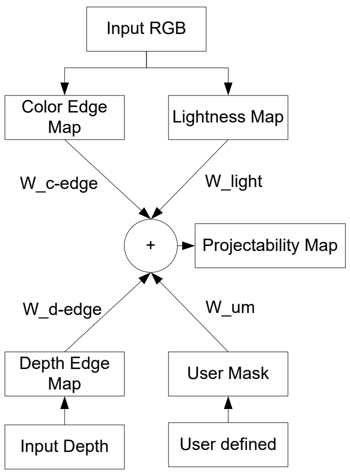


Figure 2: Abstract Structure of the projectability map calculation

Figure 3: Comparison corner detection vs. edge map: corner detection does not respond well on an image

of these approaches aim at finding large areas for projection and inherently avoid projecting on objects, as the presence of objects is used to mask the display area. As a result, they are not suitable for augmenting, for example, individual documents.

In scenarios, where the digital content should be dominant, color correction [5, 8, 17] can be used to allow uniform projection on textured surfaces. However, while color correction mitigates the problem of interference in favour of the digital content, it has some limitations – for instance printed black text on a white background cannot be made invisible.

Occlusion Management

Occlusion is a common problem on in hybrid interactive tabletops scenarios [22]. As a result, techniques have been developed to circumvent these issues by providing users with means to handle occlusion without relocating occluded items [6, 11, 13]. This is done by using interactive proxy objects, for instance icons, on free space, which allow users to access and perceive the occluded objects easily. Also, projecting content onto physical objects (instead of next to them) has been considered [20] to conserve space.

All approaches require to find areas suitable for display, whether it's finding a free spot *anywhere* on the surface to relocate the item, *next to* a physical object to display the proxy or *on* the physical object. Most systems tend to simply assume that the area right beside the "occluder" or the "margins" of an object – as they are commonly blank – are a suitable place. However, this assumption does not always hold: The surface might be dark, very textured, etc., rendering any content displayed unreadable. The messy tabletops concept [4] allows finding a large free area on back-projected tabletop. However, this approach only considers back-projected tabletops and neither provides information for top-projection setups, nor does it take into account the objects' surface.

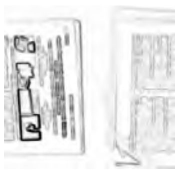
Augmented Reality

For AR-based labelling or text-display from a distance, several approaches exist: Grasset et al. [7] developed an image-based approach to avoid displaying over important regions. They employ a canny edge detector combined with a thresholded saliency map (leading to a 3-level importance map) to classify the visible area, looking for areas without edges and a low saliency. This approach however limits the insight on readability of the overlay by using a binary edge detection and only focusing on importance of the content. However, a bright white (or dark black, which is unsuitable for projection) surface has a low saliency and no edges, but the overlay won't be easy to read, which cannot be detected by this approach.

In a similar direction, Orlosky et al. [18] developed another approach to locate dark, billboard like structures in an image to overlay text. The method employs the pixel brightness and its standard deviation over the size of the object to be displayed to ensure uniformity. Hough lines algorithm is used to ensure stability, however it is not used for projectability calculation. Using standard deviation as feature is somewhat problematic, as it does not take into account the structure. A rectangle containing half black and white pixels will always have the same standard deviation, whether the upper half is black and the lower white or the pixels are randomly distributed. However, there is certainly a difference when it comes to readability between these two.

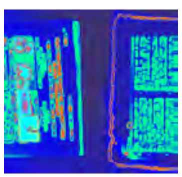
Based on preknown or generated 3d-models, work has emerged to inform the layout of AR environments. For instance, to keep track of empty spaces, rectangular ar-

(a) color image



(b) color edges

(c) depth edges



(d) final map

(e) Perspective shot of the scene

Figure 4: Physical edges are not always recognized on the color image

eas, which denote covered and uncovered areas can be used [1]. In the context of RGBD-cameras, an approach for spatial constancy has been developed [3], using a Kinect fusion based model together with a saliency map. However, both approaches require to generate a rather complex 3d-model of the scene.

All approaches except [3] neglect the physical 3d-structure of the surfaces, which is fine for augmentation from a distance. However, for close-range augmentation on physical objects, especially interactive projections, the surface structure is very important.

FreeTop's Projectability Map

To address the limitations of previous approaches and allow a more flexible and appropriate placement of projected content, we present a method to assess the visible surface in a scene regarding its suitability for projection. Instead of doing this in a binary way (projectable/not projectable), we assign a *projectability score* to each pixel. The score ranges from 0 (very suitable) to 255 (unsuitable). Areas usable for projection are, in general, of *light color* (or dark for AR-Overlays) and ideally *smooth* without any texture or physical content (e.g. text). We therefore consider lightness and smoothness of the surface, similar to [23, 24]. If the display should be interactive, a *physically continuous* and smooth surface is also desirable. FreeTop takes these factors into account and allows flexible weighting of the different aspects. This allows for more refined and flexible placement rules (e.g. it might be acceptable to have a worse projection quality, if the projection is closer to the originally intended place). As depth cameras usually provide their intrinsic calibration data, only calibration of the output device (e.g. projector) relative to the camera is needed and has to be maintained by the high-level application. To account for variability in brightness/contrast or ambient light, FreeTop employs automatic white-balance and brightness correction.

As depicted in figure 2, FreeTop generates four maps, described in the following, which are then weighted and added. The actual weights are derived from user specified weights ranging from 0 to 100 for each map by normalizing their sum to 1. This leads to a final projectability map.

1. Smoothness To locate any obstacles like text, figures etc., we choose an edge detector over the common corner detection [15]. While corners are suitable features for printed text, other content can be problematic (see figure 3 left): shapes without prominent corners, like lines, circles, etc. cannot be captured by a corner detection approach. As a result, these areas would be considered suitable for displaying content, while they are actually not.

In contrast, an edge detector does not only captures all shapes that corner detection responds to, but is additionally able to capture lines, circles and other shapes without corners (see figure 3 for a comparison). The output of the edge detection is used as *color edge map*. Conventionally, edge detectors, like Canny (e.g. used in [7]), are binary (i.e. return 1 if there is an edge and 0 if there isn't). However, as we target a continuous assessment of projectability (e.g. an edge between a yellow and a white surface is not as bad as a hard black/white contrast), we use a gradient magnitude based approach, that allows for a continuous response for edges. The gradient is derived using the Sobel operator in x and y direction, the magnitude is then the length of the resulting vector.

2. Lightness Even though the edge based approach is able to capture a wide range of possible obstacles, there is still a problem with (very) dark (or white for AR), yet

Figure 5: Example of dark areas, not recognized by edge detection

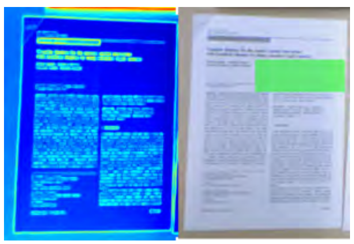


Figure 6: Largest projectable rectangle on the projectability map

smooth, surfaces that do not contain any edges (see figure 5). As mentioned, projection is more suitable on lighter surfaces. We therefore consider the surface lightness in order to find suitable spaces in form of the *lightness map*. It is defined as the inverse of the normalized and thresholded grayscale version of the color image.

3. Physical Surface In addition to surface smoothness and lightness based on the color image, we also factor in the physical surface. This is important, as the user has a different viewpoint than the camera. Hence, the surface might seem flat and continuous (i.e. no visible edges) from the camera's perspective, but from the user's perspective, the surface is, in fact, not continuous (see figure 5). Therefore, FreeTop also considers the depth image to compute a *depth edge map*, containing the physical edges of objects. Similar to the *lightness map*, a gradient magnitude based approach is used in order to factor in the height of edges. This is also important for interactive projections, as dragging for instance across a physical edge using touch is obviously impractical [19].

4. User Mask Besides projection quality aspects, an application or the user might have additional personalized constraints (e.g. masking a certain area to avoid projection). These constraints can be considered in form of a *user mask*, allowing to specify bias values for certain areas. For example, if no projection should occur within the users working area, the mask would assign the area a high penalty value. It can be either specified by directly drawing on the camera image or by more advanced techniques implemented by the application using FreeTop (e.g. allowing the use of an actual brush on the physical world).

Operating Modes

To support different scenarios, FreeTop can operate in two different modes, depending on the requirements:

Dynamic mode: In this mode, the whole camera image is analyzed in every frame. This mode is suitable in scenarios where no or only very few specific objects are tracked individually, or in highly dynamic scenes (e.g. many users are physically drawing on a large digitally augmented map). Also, in mobile ad-hoc scenarios, this method is favorable, as the present objects are not known upfront. As the map is updated in real-time, moving objects or changing surface textures do not cause any issues. One possible issue in this mode is the interference with the projected content. However, there are generally two possible scenarios: 1) The content which should be projected at a place is currently projected there, then, it is known that it was projected there before and the low projectability score is due to the current projection. Hence, it is safe to project there again, as projectability has been assessed before initially projecting. 2) The content is new and not yet projected. Then, the interference is useful, as it invalidates the area currently used for projection for further projection (projections should not overlap).

Static mode: If used in applications where objects in the projection area are tracked by the system, FreeTop can be used in *static mode*. In this mode, projectability maps are computed upfront and remain static during the use of the system. This mitigates any possible interference between the map computation and projection. The computation of the projectability map can be decomposed into a background part (the empty working area) and individual parts for the objects, with the overall map being constructed in real-time based on the object tracking data. As a result, the analysis can be done on high-res images for the visual maps and, with better depth cams in the future, also for the depth edge map, as it is only done once. Additionally, the maps for objects can be generated from the digital

Figure 7: Highlighted projectable areas (green)

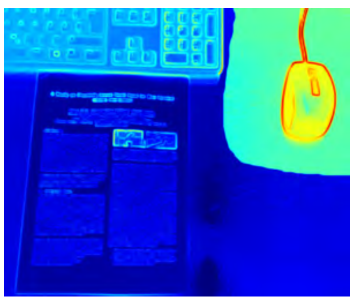


Figure 8: User Mask avoids projection on mouse area by adding penalty

version (e.g. for a printed document). Also, it allows for better run-time performance as there are no runtime computations. Finally, the user can modify the projectability map for individual objects, e.g. through an application's UI as they are available "offline".

Layout with FreeTop

Layout based on FreeTop can be done in several ways: The most trivial one is to search the largest rectangle with a score below a threshold and then display all content within this rectangle (fig. 6). However, more flexibility can be achieved by taking all areas with a score below a (dynamic) threshold into account (see figure 7, note that projection across the object borders is avoided). In case there are any soft preferences regarding the placement, the user mask can be used to accommodate them by providing a higher penalty for locations in which projection is unwanted (see figure 8, here the user prefers to have no projection around the mouse). Thereby, the masked areas are not fully ignored, but only used if there is no better suitable area. The use of a mask instead of incorporating it into the layout algorithm is beneficial as it reduces the complexity of the layout step while the user can for instance intuitively "paint" the areas he does not want projection to happen.
Another possibility is to perform advanced layout based on the original map, like a gradient based optimization approach where the object is moved along the gradient of the map to a suitable (local) minimum.

System Description and Performance

We implemented the FreeTop concept using C++, OpenCV and a Microsoft Kinect. As a proof of concept, we search the largest available rectangle assessed suitable for projection using the dynamic mode described before (i.e. no caching, fully processing every frame). We ran FreeTop on a Linux PC with an Intel i5 processor. The OpenCV GPU acceleration was not used, as it is not available on every hardware. Frame time was measured over map computation and rectangle finding across 100 frames.

At Full-HD resolution, the average frame processing time is about 74 ms (13.5 fps, independent on scene contents) which is suitable for projectability assessment in low-dynamic scenes (e.g. a desk). For scenes with higher dynamic, a lower resolution can be used, allowing up to 60 fps at 640×480. Through parallelization or GPU acceleration, the performance can be further improved.

Summary

We present FreeTop, an approach to free-spot identification in arbitrary environments. It can be used for on-surface augmentation using projection or AR-glasses. Contrary to existing approaches [2, 15], it does not rely on a single feature, but uses a multitude of factors (lightness, smoothness, physical surface as well as user preferences) to assess projectability on a continuous scale. This allows to flexibly adapt projection to the situation, as it is possible to respect trade-offs (e.g. closeness to the intended position vs. projection quality). Being based on a standard depth camera assembly, which is available off-the-shelf and integrated in modern AR-glasses, FreeTop can easily be used in stationary and mobile settings. As a next step, we aim to explore FreeTops use in mobile scenarios more deeply.

Acknowledgements

This work was funded by the German Federal Ministry of Education and Research (BMBF) [01IS12054] and the by the IITP grant funded by the Korean government (MSIP) [B0101-15-1292, Development of Smart Space to promote the Immersive Screen Media Service]

References

[1] Blaine Bell, Steven Feiner, and Tobias Höllerer. 2001. View Management for Virtual and Augmented Reality. In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology (UIST '01)*. ACM, New York, NY, USA, 101–110. DOI: http://dx.doi.org/10.1145/502348.502363

[2] Daniel Cotting and Markus Gross. 2006. Interactive environment-aware display bubbles. In *Proceedings of the 19th annual ACM symposium on User interface software and technology (UIST '06)*. ACM, New York, NY, USA, 245–254. DOI:http://dx.doi.org/10.1145/1166253.1166291

[3] Barrett Ens, Eyal Ofek, Neil Bruce, and Pourang Irani. 2015. Spatial Constancy of Surface-Embedded Layouts Across Multiple Environments. In *Proceedings of the 3rd ACM Symposium on Spatial User Interaction (SUI '15)*. ACM, New York, NY, USA, 65–68. DOI:http://dx.doi.org/10.1145/2788940.2788954

[4] Euan Freeman and Stephen Brewster. 2013. Messy Tabletops: Clearing Up the Occlusion Problem. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13)*. ACM, New York, NY, USA, 1515–1520. DOI:http://dx.doi.org/10.1145/2468356.2468627

[5] Kensaku Fujii, Michael D. Grossberg, and Shree K. Nayar. 2005. A ProjectorCamera System with Real-Time Photometric Adaptation for Dynamic Environments. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, Vol. 2. IEEE Computer Society, Washington, DC, USA, 1180 vol. 2–. DOI:http://dx.doi.org/10.1109/CVPR.2005.42

[6] Genki Furumi, Daisuke Sakamoto, and Takeo Igarashi. 2012. SnapRail: a tabletop user interface widget for addressing occlusion by physical objects. In *Proceedings of the 2012 ACM international conference on Interactive tabletops and surfaces (ITS '12)*. ACM, New York, NY, USA, 193–196. DOI:http://dx.doi.org/10.1145/2396636.2396666

[7] Raphael Grasset, Tobias Langlotz, Denis Kalkofen, Markus Tatzgern, and Dieter Schmalstieg. 2012. Image-driven View Management for Augmented Reality Browsers. In *Proceedings of the 2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR) (ISMAR '12)*. IEEE Computer Society, Washington, DC, USA, 177–186. DOI:http://dx.doi.org/10.1109/ISMAR.2012.6402555

[8] M.D. Grossberg, H. Peri, S.K. Nayar, and P.N. Belhumeur. 2004. Making one object look like another: controlling appearance using a projector-camera system. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, Vol. 1. IEEE Computer Society, Washington, DC, USA, I–452–I–459 Vol.1. DOI: http://dx.doi.org/10.1109/CVPR.2004.1315067

[9] Björn Hartmann, Meredith Ringel Morris, Hrvoje Benko, and Andrew D. Wilson. 2010. Pictionaire: Supporting Collaborative Design Work by Integrating Physical and Digital Artifacts. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work (CSCW '10)*. ACM, New York, NY, USA, 421–424. DOI: http://dx.doi.org/10.1145/1718918.1718989

[10] Jochen Huber, Jürgen Steimle, Chunyuan Liao, Qiong Liu, and Max Mühlhäuser. 2012. LightBeam: Interacting with Augmented Real-world Objects in Pico Projections. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia (MUM '12)*. ACM, New York, NY, USA, Article 16, 10 pages. DOI:http://dx.doi.org/10.1145/2406367.2406388

[11] Waqas Javed, KyungTae Kim, Sohaib Ghani, and Niklas Elmqvist. 2011. Evaluating Physical/Virtual Occlusion Management Techniques for Horizontal Displays. In *Human-Computer Interaction - INTERACT 2011*, Pedro Campos, Nicholas Graham, Joaquim Jorge, Nuno Nunes, Philippe Palanque, and Marco Winckler (Eds.). Lecture Notes in Computer Science, Vol. 6948. Springer Berlin Heidelberg, Heidelberg, Germany, 391–408. DOI: http://dx.doi.org/10.1007/978-3-642-23765-2_27

[12] Shaun K. Kane, Daniel Avrahami, Jacob O. Wobbrock, Beverly Harrison, Adam D. Rea, Matthai Philipose, and Anthony LaMarca. 2009. Bonfire: A Nomadic System for Hybrid Laptop-tabletop Interaction. In *Proceedings of the 22Nd Annual ACM Symposium on User Interface Software and Technology (UIST '09)*. ACM, New York, NY, USA, 129–138. DOI:http://dx.doi.org/10.1145/1622176.1622202

[13] Mohammadreza Khalilbeigi, Jürgen Steimle, Jan Riemann, Niloofar Dezfuli, Max Mühlhäuser, and James D. Hollan. 2013. ObjecTop: occlusion awareness of physical objects on interactive tabletops. In *Proceedings of the 2013 ACM international conference on Interactive tabletops and surfaces (ITS '13)*. ACM, New York, NY, USA, 255–264. DOI:http://dx.doi.org/10.1145/2512349.2512806

[14] H. Koike, Y. Sato, and Y. Kobayashi. 2001. Integrating paper and digital information on EnhancedDesk: a method for realtime finger tracking on an augmented desk system. *ACM Trans. Comput.-Hum. Interact.* 8, 4 (Dec. 2001), 307–322. DOI:http://dx.doi.org/10.1145/504704.504706

[15] Chunyuan Liao, Hao Tang, Qiong Liu, Patrick Chiu, and Francine Chen. 2010. FACT: fine-grained cross-media interaction with documents via a portable hybrid paper-laptop interface. In *Proceedings of the international conference on Multimedia (MM '10)*. ACM, New York, NY, USA, 361–370. DOI:http://dx.doi.org/10.1145/1873951.1874001

[16] H. Mitsuhara, Y. Yano, and T. Moriyama. 2010. Paper-top interface for supporting note-taking and its preliminary experiment. In *Systems Man and Cybernetics (SMC), 2010 IEEE International Conference on*. IEEE Computer Society, Washington, DC, USA, 3456–3462. DOI:http://dx.doi.org/10.1109/ICSMC.2010.5642448

[17] Shree K. Nayar, Harish Peri, Michael D. Grossberg, and Peter N. Belhumeur. 2003. A projection system with radiometric compensation for screen imperfections. In *IEEE ICCV Workshop on Projector-Camera Systems (PROCAMS)*. IEEE Computer Society, Washington, DC, USA, 8.

[18] Jason Orlosky, Kiyoshi Kiyokawa, and Haruo Takemura. 2013. Dynamic Text Management for See-through Wearable and Heads-up Display Systems. In *Proceedings of the 2013 International Conference on Intelligent User Interfaces (IUI '13)*. ACM, New York, NY, USA, 363–370. DOI:http://dx.doi.org/10.1145/2449396.2449443

[19] Jan Riemann, Mohammadreza Khalilbeigi, and Max Mühlhäuser. 2015. In-Situ Occlusion Resolution for Hybrid Tabletop Environments. In *Human-Computer Interaction – INTERACT 2015*, Julio Abascal, Simone Barbosa, Mirko Fetter, Tom Gross, Philippe Palanque, and Marco Winckler (Eds.). Lecture Notes in Computer Science, Vol. 9298. Springer International Publishing, Cham, Switzerland, 278–295. DOI:http://dx.doi.org/10.1007/978-3-319-22698-9_18

[20] Jan Riemann, Mohammadreza Khalilbeigi, and Max Mühlhäuser. 2013. PeriTop: extending back-projected tabletops with top-projected peripheral displays. In *Proceedings of the 2013 ACM international conference on Interactive tabletops and surfaces (ITS '13)*. ACM, New York, NY, USA, 349–352. DOI:http://dx.doi.org/10.1145/2512349.2512397

[21] Thitirat Siriborvornratanakul and Masanori Sugimoto. 2008. Clutter-aware Adaptive Projection Inside a Dynamic Environment. In *Proceedings of the 2008 ACM Symposium on Virtual Reality Software and Technology (VRST '08)*. ACM, New York, NY, USA, 241–242. DOI:http://dx.doi.org/10.1145/1450579.1450633

[22] Jürgen Steimle, Mohammadreza Khalilbeigi, Max Mühlhäuser, and James D. Hollan. 2010. Physical and digital media usage patterns on interactive tabletop surfaces. In *ACM International Conference on Interactive Tabletops and Surfaces (ITS '10)*. ACM, New York, NY, USA, 167–176. DOI:http://dx.doi.org/10.1145/1936652.1936685

[23] K. Tanaka, Y. Kishino, M. Miyamae, T. Terada, and S. Nishio. 2007. An Information Layout Method for an Optical See-through HMD Considering the Background. In *Wearable Computers, 2007 11th IEEE International Symposium on*. IEEE Computer Society, Washington, DC, USA, 109–110. DOI:http://dx.doi.org/10.1109/ISWC.2007.4373791

[24] Vineet Thanedar and Tobias Höllerer. 2004. *Semi-automated Placement of Annotations in Videos*. Technical Report 2004-11. UC, Santa Barbara.

[25] Pierre Wellner. 1993. Interacting with Paper on the DigitalDesk. *Commun. ACM* 36, 7 (July 1993), 87–96. DOI:http://dx.doi.org/10.1145/159544.159630

[26] Robert Xiao, Chris Harrison, and Scott E. Hudson. 2013. WorldKit: Rapid and Easy Creation of Ad-hoc Interactive Applications on Everyday Surfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 879–888. DOI:http://dx.doi.org/10.1145/2470654.2466113