








# Enabling Collaborative Cybercartography with MapBlender

Marius Hogräfer <sup>1</sup>, Jens Emil Grønbæk <sup>1</sup>, Jana Puschmann <sup>1</sup>, Sebastian Krog Knudsen <sup>1</sup>,  
and Hans-Jörg Schulz <sup>1</sup>

<sup>1</sup>Department of Computer Science, Aarhus University, Denmark

Correspondence: Marius Hogräfer ([mhograefer@cs.au.dk](mailto:mhograefer@cs.au.dk))

**Abstract.** One of the challenges that GIS users in diverse, distributed teams face these days is being able to efficiently collaborate, both across workspaces and tools. To that end, we present *MapBlender*, a cybercartographic application that fosters geocollaboration by adapting a collaboration-first approach, placing users and their GIS tools on equal footing. *MapBlender* allows all connected users to share video feeds of their GIS tools and webcams, which can then be freely rearranged and re-sized, adopting simple and familiar interaction techniques from modern window managers. In addition, the rendering and translucency of these feeds can be adjusted, allowing users to align and analyze information across tools. We demonstrate the utility of *MapBlender* in a hybrid teaching scenario, wherein a representative of a software company uses our application to instruct co-located and remote learners from another company on the features of their GIS tool. *MapBlender* is publicly available under open source licenses and runs in the browser, with no local installation necessary. Thus, with *MapBlender* in place, one of our goals is to promote an HCI perspective on future work in geocollaboration.

**Keywords.** cybercartography, collaboration, GIS

## 1 Introduction

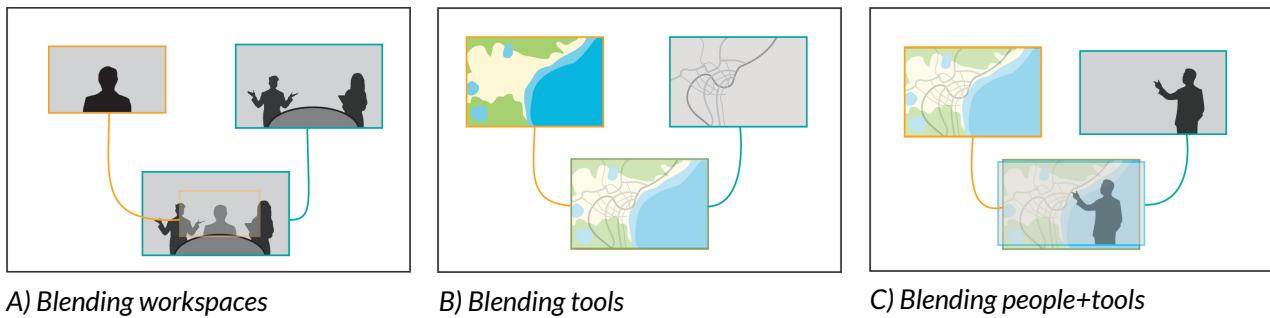
Taylor’s vision of Cybercartography bringing together various kinds of spatial information on interactive, multimedia interfaces is 25 years old as of this year (Taylor, 1997). Yet in practice, incompatibilities, licensing issues, or simply working in a distributed team still prevent us from sharing, recombining, and otherwise simply engaging with different spatial information in a joint fashion. While research fields like *collaborative GIS* (Balram and Dragicevic, 2006) or *geocollaboration* (Maceachren and Brewer, 2004) have made great progress on bridging the gaps between data sources, tools, and users, it seems that “we are still not there yet.” This motivates a very active development of collaborative GIS tools – for example, the

systems presented by Aguilar et al. (2020) or Yaqin et al. (2022), to name just two recent ones.

It is hard to attribute this lack of adoption to a single cause. In their review of real-time collaborative GIS systems, Sun and Li (2016) point towards a number of open questions and missing features that may underlie this problem. Among their list, we have identified the need for multi-platform support (from tabletops to tablets) and open source availability to be two main hurdles. In addition, there is also a third point that we observed in previous work on co-located collaborative cybercartography tools (Eichner et al., 2015): ease of use and a low learning curve. This is particularly important for casual users and non-experts, such as policy makers or project economists who need to be brought into collaborative geospatial decision processes with minimal effort.

As a result of this analysis and a solution to the above mentioned hurdles, we present the free web app *MapBlender* that aims to alleviate these issues by providing a simple and seamless way to “blend” outputs across tools, devices, and users. To that end, *MapBlender* enables a layering of tools atop each other in a similar way as combining cartographic layers to generate a map. Thanks to the use of an underlying video-conferencing platform with functionality for sharing and layering multiple screens (Grønbæk et al., 2021), these layers of tool windows can even come from different devices at different locations. As a result, *MapBlender* can bring together different users with different expertise, different tool sets, different data sources from different locations to work and learn collaboratively.

In the following, we detail *MapBlender* – its features, implementation, and availability – as well as a scenario illustrating its professional use for in a hybrid teaching / tutorial scenario.



**Figure 1.** Three ways in which *MapBlender* enables collaboration through blending.

## 2 *MapBlender*: Enabling geocollaboration across platforms

In this section, we introduce *MapBlender* along its design requirements, features, and how the interface supports them, as well as going into some implementation details.

### 2.1 Design Requirements

Our main aim is to provide a software platform that can seamlessly serve as the “glue” between different cartographic tools, device platforms, and distributed workspaces, in order to facilitate lightweight coordination and collaboration (Schulz et al., 2019). This can be achieved in different ways depending on the particular requirements of the tool context. For the case of *MapBlender*, we identify the following three design requirements:

- DR1: Easy to learn and use** The software application should use familiar (i.e., well-established, rather than novel) interaction and interface concepts, such that it is easy to learn and to use for most users of GIS and video conferencing software.
- DR2: Multi-platform support** The software should work across any computing platform to allow people with different setups (screen spaces, operating systems, input modalities, etc.) to work together.
- DR3: Open source** The software should be accessible to the community for extension and modification to support the specific needs of different cartographic collaborative work practices.

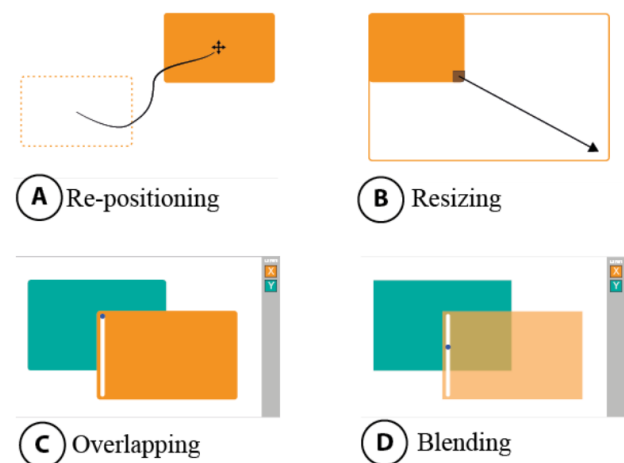
### 2.2 Features

In general, *MapBlender* facilitates geocollaboration, by allowing users to share live video streams of their GIS tools, and live streams of their webcams including live audio. These video feeds are visible to all other users in a session, similar to any general purpose video conferencing tool, and can be freely arranged by all users on a shared workspace. This setup makes *MapBlender* highly flexible in its usage: Which tools are shared in those video feeds, how many tools are shared at the same time, what con-

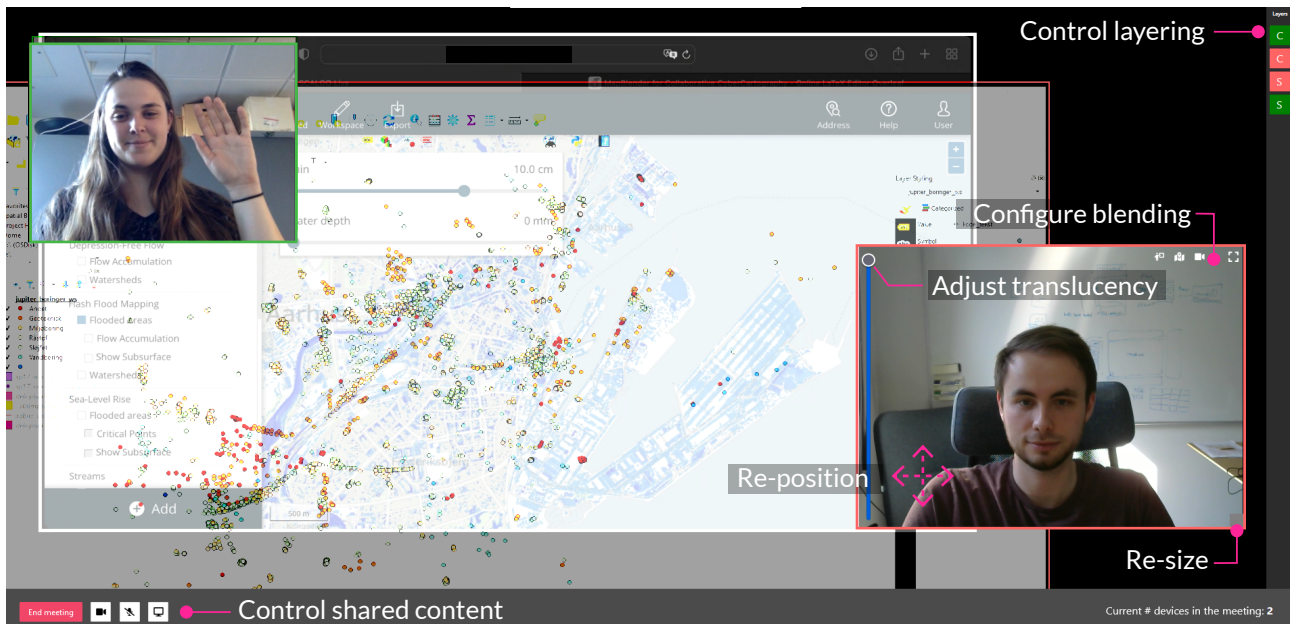
tent the webcam is recording, as well as the way the video feeds are arranged can be freely negotiated by the users.

Additionally, *MapBlender* goes beyond the goals of video conferencing platforms, by providing dedicated collaboration features for the GIS context. In particular, our application adapts the layering metaphor, which is normally used to relate different kinds of thematic data with each other in their geospatial context on one map in one GIS tool (Dent et al., 2008, pp. 209). In *MapBlender*, we apply layering to the tool level by placing video feeds of the respective tools atop of each other. This way, it becomes possible to layer the information from multiple maps across multiple tools. Users can then adjust the way these windows are *blended* with each other, for example by changing their translucency, their order, or the way in which the renderer combines their contents. Below, we explain *MapBlender*’s features in more detail, looking at how each addresses the design requirements we introduced in the previous section. [Figure 1](#) gives an overview on how collaboration is supported in *MapBlender*.

**Interaction Techniques** To facilitate a low learning curve (DR1), *MapBlender* is built around a small set of simple interaction techniques, that are widely used across all modern window managers (see illustration in [Figure 2](#)). In



**Figure 2.** Interactions with shared content in *MapBlender*. Reproduced with permission from Grønbaek et al. (2021).



**Figure 3.** Screenshot of *MapBlender* interface. It shows two users collaborating in a session, with both sharing their webcams and GIS tools, in particular QGIS and SCALGO live. Translucent window management enables cross-tool collaboration: Using blend modes, information seen in both tools becomes visible at once, and users can manually reposition windows to align the visualized data.

particular, all video feeds are shown as windows that can be freely re-positioned and re-sized via direct manipulation. To re-position, users can drag the window to the new location, and to re-size, they can drag on the bottom right corner of each window. The order of views is shown as a simple stack on the right side of the interface, which can be reordered using drag-and-drop. Through this small and well-defined set of interaction possibilities, *MapBlender* reduces usage barriers (DR1) and increases cross-platform compatibility (DR2), leaving specialized interactions to specialized GIS tools.

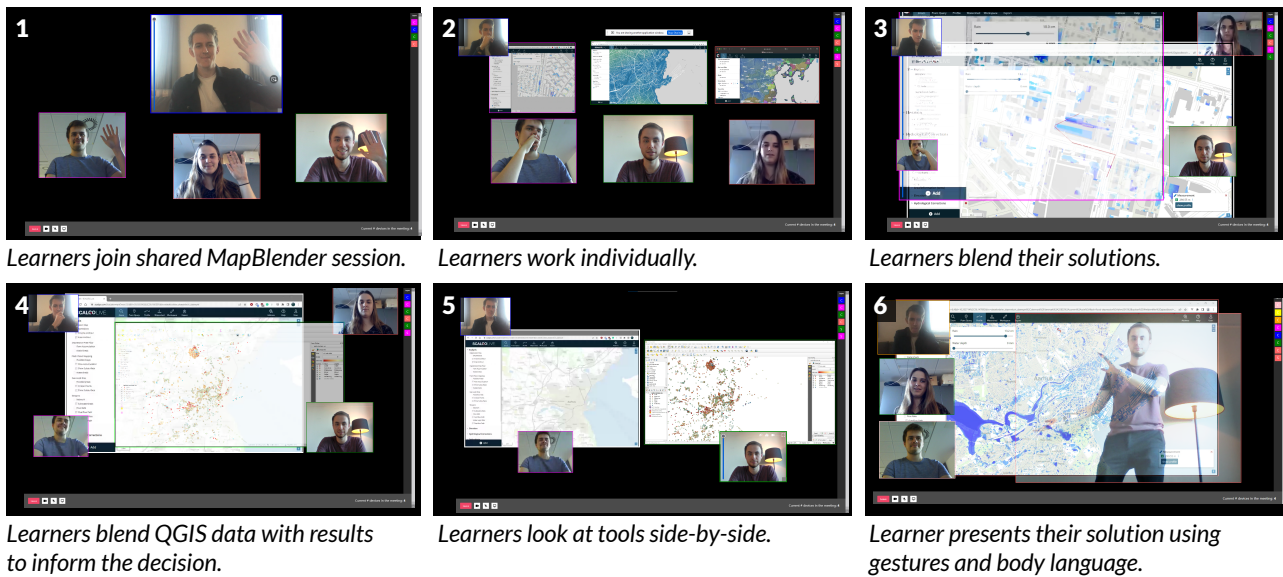
**WYSIWIS workspace** To further facilitate ease of use (DR1), *MapBlender* follows the “What You See Is What I See” principle (WYSIWIS) (Stefik et al., 1987), i.e., the window layout appears identical to all users that join the session, meaning that changes that one makes to this layout are propagated immediately to all other clients as well. This entails that all users have a common frame of reference when referring to parts of the interface, for instance when discussing parts on the map, to guide new users towards particular features, or when asking questions about the tools. Since tools are shared in *MapBlender* as video streams, they can only be controlled by the user that shared them with the group. Thus, the workspace that everybody can see automatically (and implicitly) serves as a *shared* workspace, while the interfaces of the tools themselves running on users’ local machines serve as *private* workspaces, thus allowing for collaborative awareness (Sun and Li, 2016). Moreover, users can place themselves inside different shared screens to point and make reference to insights in the visualizations.

**Translucent Window Management** The core feature of *MapBlender* is that it allows to easily blend the information from multiple GIS tools and webcam feeds with each other. An overview of *MapBlender*’s interface is depicted in Figure 3. To that end, *MapBlender* provides options for managing the translucency and appearance of video feeds, that is, controlling the visibility of window contents. When placing tool windows atop of each other, users can reduce then the transparency of each window by increasing its transparency and vice-versa, using the slider on the left side of that window. For cases where transparency is not enough, *MapBlender* allows further adjustments to the video presentation. Our application provides three presets, each supplying different adjustments for particular use cases:

- *tool mode*: increases the contrast and brightness of a video feed to make lines and points in GIS tools more visible, for layering shared screens)
- *camera mode*: improves the rendering of users on-top of shared screens, by turning the camera feed to grayscale, thus eliminating colors from the webcam)
- *presenter mode*: improves the visibility of users’ gestures to point towards content in shared screens)

### 2.3 Implementation Details

In order to ensure platform independence (DR2), *MapBlender* is built using basic web technologies (i.e., JavaScript, HTML, and CSS), which also means that it can be started from the browser, such that no local installation is necessary (DR1). Our application is an extension of MirrorBlender (Grønbaek et al., 2021), a gen-



**Figure 4.** Illustration of the use case discussed in section 3. A teacher instructs three learners at a company on the features of a new GIS tool using *MapBlender*.

eral purpose video conferencing system. MirrorBlender provides the basic collaboration functionalities we use in *MapBlender*, namely the WYSIWIS window management for live webcam feeds and shared screens, including the ability to adjust the window transparency. MirrorBlender itself is built using the Webstrates environment (Klok-mose et al., 2015) providing the low-level synchronization of contents between users, and Codestrates (Rädle et al., 2017; Borowski et al., 2021), a live code editor for Webstrate pages. The former makes our application easily accessible (DR1) via a simple link that users open in their browsers (created by the link generator <https://demo.webstrates.net/mapblender-release?copy><sup>1</sup>). It also allows users to essentially share any local GIS tool with other users, thus providing platform independence on the tool level (DR2). The Codestrates editor, on the other hand, allows users to customize the appearance and functionality of *MapBlender* to the needs of their particular use case. This malleability principle is quite powerful; in fact, *MapBlender* is built by extending a fork of MirrorBlender for the GIS context exclusively using the Codestrates editor. An added benefit of this technology stack is that the source code to all components of *MapBlender* is openly accessible (DR3).

### 3 Use Case: Facilitating Distributed Cybercartographic Tutorials with *MapBlender*

Next, we demonstrate how the features of *MapBlender* enable geocollaboration. We do so in the context of software training, where an educator introduces a group of

course participants to the features of GIS tools, using practical exercises. This use case is based on the experiences we gained in collaboration with SCALGO<sup>2</sup>, a local company that develops and distributes the flood prevention planning tool SCALGO live to customers in Northern Europe. Part of their offerings are training sessions for new customers of their software, which take place in hybrid meetings, where some participants are located in the same room as the teacher, while others join in from remote locations. As part of these trainings, participants often need to solve tasks that integrate other tools like QGIS 3<sup>3</sup> with that new software, which prepares participants for a common workflow when using that tool in their work. A coach from the company reported that integrating remote participants through established video conferencing tools is challenging, as working “side-by-side” with them is hindered. However, with *MapBlender*, they can simply share live feeds of their tools and webcams with each other and collaborate on the shared workspace, using blending to accomplish their tasks. The following use case is illustrated in Figure 4.

We consider a use case, in which three learners (L1, L2, and L3) working for a municipality collaborate on a shared exercise around SCALGO live, in a course hosted by teacher T in a meeting room at the office of a municipality. L1 and L2 join the session remotely, while L3 sits in the meeting room, together with T and other course participants. Remote participants join the session via a *MapBlender* meeting room set up by T, and watch as she demonstrates how to identify hazardous watersheds caused by flash flood events with SCALGO live, and how to prevent them from forming by simulating drainages.

<sup>1</sup>Clicking this link generates a unique URL with a copy of the original release version. In order to start a meeting, participants need to share the newly generated URL.

<sup>2</sup><https://scalgo.com>

<sup>3</sup><https://www.qgis.org/en/site/index.html>

Co-located participants see this projected on a large display in the room.

For the practical part of the tutorial, T asks participants to solve an exercise, in which they should propose a way to identify watersheds in a dense residential area, and then export their solution to QGIS. L1, L2, and L3 decide to join forces and open a separate meeting room on *MapBlender*, which they all join from their laptops, sharing their webcam feeds. They familiarize themselves with the interface, each learner working independently on the exercise at first, to later compare solutions.

After the group individually solved the exercise, they move on to discuss their solutions by sharing their screens in *MapBlender*. They arrange their shared screens using drag and drop, resizing them to fit the entire workspace. To increase the visibility of the area polygons representing watersheds, L2 adjusts the transparencies and activates the “map” preset for each shared screen. When blending their solutions in this way, similarities and differences quickly become apparent. The learners notice that, while they identified the same hazardous watershed polygons, they found different ways of draining them. To inform their discussion, L3 opens up a map he created for his work at the municipality in QGIS, i.e., data to which SCALGO live has no access. By blending this map with their solutions, they quickly agree on a solution that drains the watershed into a close-by street drainage with high capacity that the municipality plans to construct. They export their solution from SCALGO live to QGIS and save it.

In the next part of the lesson, the remote learners join the session hosted by T again. L2 demonstrates the group’s solution to T and the other participants through *MapBlender*, overlaying his webcam feed on a shared QGIS view. L2 activates the “presenter” preset, which increases his visibility when overlaid on GIS tools. He also steps away from the desk in his office and positions himself in front of a white wall, which in the interface blends his body with the interface of their GIS tools, similar to a weather reporter. While participating in the training remotely, *MapBlender* allows L2 to use gestures to point at interesting parts of the group’s solution, and to use his body language while presenting. After L2 finished presenting, T shares his prepared solution, again directly in their shared *MapBlender* session.

#### 4 Discussion

Here, we discuss the *MapBlender* in terms of the application criteria: novelty, usability, communication efficiency, generalizability, and current limitations.

**Novelty:** Having grown out of an HCI research project, *MapBlender* follows a unique approach in which users and applications are on equal footing. Where other real-time geocollaboration tools center on the GIS applications and reduce the collaboration between users to an added widget

on the side – e.g., a chat window (Hall and Leahy, 2008), *MapBlender* puts an emphasis on user collaboration and live interaction that is hardly matched by any of the existing systems.

**Usability:** *MapBlender* inherits many of its functionalities from video conferencing platforms, which are already widely used to foster collaboration across domains in both professional and casual contexts, with Zoom, Microsoft Teams, and Google Hangouts being some popular examples. As a result, the basic features of *MapBlender* pose a low entry barrier, as they should already be familiar to most first-time users. GIS-specific features like adjusting the blend mode of windows to increase the readability of map symbols are extensions of this video conferencing concept, such that *MapBlender* already supports collaboration without them, and users can gradually include them in their work when needed.

**Communication Efficiency:** The interaction principles of *MapBlender* provide the familiarity of traditional OS window management with the additional capabilities of blending tool windows (i.e., translucency slider, adjusting the ordering, and presets for blending maps vs. people). This unique combination provides a method for cartography that aligns well with the principle of cartographic layering, thus lowering the learning curve and minimizing the need for lengthy explanations or tutorials.

**Generalizability:** Since *MapBlender* is agnostic of the underlying tools being used and blended with it, it caters to a wide variety of use cases, depending on the tools being brought in. Its foundation in *Codestrates* and *Webstrates* provides the possibility for customization and extension if necessary.

**Limitations:** While being available online to anyone interested in using it, *MapBlender* is far from a full-fledged product. At the time of this writing, *MapBlender* is, for instance, still limited in the following regards:

- *MapBlender* only supports about a handful of participants at a time. This is due to the lack of a central server, which is why the current video sharing in the underlying *MirrorBlender* platform requires video streaming between each pair of participants.
- There is currently no support for blending maps using different projections, such as a “snap-to-map” functionality that would try to automatically sync blended maps with each other.
- For now, the *MapBlender* prototype is developed and tested for Chromium-based browsers, which means that some features and UI may not work as expected in other browsers. However, these issues can be solved in future iterations given more development resources.

## 5 Conclusions and Future Work

In this paper, we presented *MapBlender*, a cybercartographic application that fosters collaboration by allowing users to blend video feeds of their respective cartographic tools, as well as live feeds from their webcams in a shared workspace. We demonstrated the utility of our solution in a use case around for teaching GIS software to professional users. The code to *MapBlender*, as well as a live demo and a demonstration video showcasing its use can be found on <https://vis-au.github.io/mapblender>.

*MapBlender* makes it simple to collaborate across GIS tools and workspaces. Beyond the professional setting we presented in this paper, its features also apply to more casual settings, such as collaboratively planning a hiking trip. People using *MapBlender* can share the interfaces of their respective travel planning tool and split the planning workload among them, with one taking the role of planning the hiking route in Kahoot, another identifying points of interest using Google Maps, and a third person finding overnight locations on AirBnB. With *MapBlender*, they can overlay the respective maps of these applications to put their work in context with each other, thus discussing their options across tools. Future iterations of *MapBlender* can investigate these use cases.

## References

- Aguilar, R., Flacke, J., and Pfeffer, K.: Towards Supporting Collaborative Spatial Planning: Conceptualization of a Mappable Tool through User Stories, *ISPRS International Journal of Geo-Information*, 9, 29:1–29:18, <https://doi.org/10.3390/ijgi9010029>, 2020.
- Balram, S. and Dragicevic, S., eds.: *Collaborative Geographic Information Systems*, Idea Group Publishing, <https://doi.org/10.4018/978-1-59140-845-1>, 2006.
- Borowski, M., Kristensen, J. B., Bagge, R., and Klokmoose, C. N.: *Codestrates v2: A Development Platform for Webstrates*, Tech. rep., Aarhus University, 2021.
- Dent, B. D., Torguson, J., and Hodler, T. W.: *Cartography: Thematic Map Design*, McGraw-Hill, 2008.
- Eichner, C., Nocke, T., Schulz, H.-J., and Schumann, H.: Interactive Presentation of Geo-Spatial Climate Data in Multi-Display Environments, *International Journal of Geo-Information*, 4, 493–514, <https://doi.org/10.3390/ijgi4020493>, 2015.
- Grønbæk, J. E., Saatçi, B., Griggio, C. F., and Klokmoose, C. N.: *MirrorBlender: Supporting Hybrid Meetings with a Malleable Video-Conferencing System*, in: Proc. of CHI, ACM, <https://doi.org/10.1145/3411764.3445698>, 2021.
- Hall, G. B. and Leahy, M. G.: Design and Implementation of a Map-Centred Synchronous Collaboration Tool Using Open Source Components: The MapChat Project, in: *Open Source Approaches in Spatial Data Handling*, edited by Hall, G. B. and Leahy, M. G., pp. 221–245, Springer, [https://doi.org/10.1007/978-3-540-74831-1\\_11](https://doi.org/10.1007/978-3-540-74831-1_11), 2008.
- Klokmoose, C. N., Eagan, J. R., Baader, S., Mackay, W., and Beaudouin-Lafon, M.: *Webstrates: Shareable Dynamic Media*, in: Proc. of the ACM Symposium on User Interface Software and Technology (UIST), pp. 280–290, ACM, <https://doi.org/10.1145/2807442.2807446>, 2015.
- Maceachren, A. M. and Brewer, I.: Developing a conceptual framework for visually-enabled geocollaboration, *International Journal of Geographical Information Science*, 18, 1–34, <https://doi.org/10.1080/13658810310001596094>, 2004.
- Rädle, R., Nouwens, M., Antonsen, K., Eagan, J. R., and Klokmoose, C. N.: *Codestrates: Literate Computing with Webstrates*, in: Proc. of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST), pp. 715–725, ACM, <https://doi.org/10.1145/3126594.3126642>, 2017.
- Schulz, H.-J., Röhlig, M., Nonnemann, L., Aehnelt, M., Diener, H., Urban, B., and Schumann, H.: *Lightweight Coordination of Multiple Independent Visual Analytics Tools*, in: Proceedings of the 10th International Conference on Information Visualization Theory and Applications (IVAPP), edited by Kerren, A., Hurter, C., and Braz, J., pp. 106–117, SciTePress, Prague, Czech Republic, <https://doi.org/10.5220/0007571101060117>, 2019.
- Stefik, M., Bobrow, D. G., Foster, G., Lanning, S., and Tatar, D.: *WYSIWIS Revised: Early Experiences with Multiuser Interfaces*, *ACM Transactions on Information Systems*, 5, 147–167, <https://doi.org/10.1145/27636.28056>, 1987.
- Sun, Y. and Li, S.: Real-time collaborative GIS: A technological review, *ISPRS Journal of Photogrammetry and Remote Sensing*, 115, 143–152, <https://doi.org/10.1016/j.isprsjprs.2015.09.011>, 2016.
- Taylor, D. R. F.: *Maps and mapping in the information era*, in: Proc. of the 18th ICA/ACI International Cartographic Conference, edited by Ottoson, L., pp. 1–10, 1997.
- Yaqin, S., Chen, X., Jinxiang, W., Hang, Y., Haibo, S., Xiumei, Z., and Lin, Z.: Development of a multi-view and geo-event-driven real-time collaborative GIS, *International Journal of Digital Earth*, 15, 134–147, <https://doi.org/10.1080/17538947.2021.2017034>, 2022.