Visualizing and Manipulating Automatic Document Orientation Methods Using Vector Fields

Pierre Dragicevic, Yuanchun Shi

HAL Id: hal-00734622
https://inria.hal.science/hal-00734622

Submitted on 24 Sep 2012

Visualizing and Manipulating Automatic Document Orientation Methods Using Vector Fields

Pierre Dragicevic [1,2]
(1) INRIA
France
dragice@lri.fr

Yuanchun Shi [2]
(2) Key Laboratory of Pervasive Computing
Tsinghua University, China
shiyc@tsinghua.edu.cn

ABSTRACT

We introduce and illustrate a design framework whereby tabletop documents are oriented according to vector fields that can be visualized and altered by end users. We explore and illustrate the design space using interactive 2D mockups and show how this approach can potentially combine the advantages of the fully manual and fully automatic document orientation methods previously proposed in the literature.

Author Keywords

Tabletop, orientation, vector fields.

ACM Classification Keywords

H5.m. Information interfaces and presentation.

INTRODUCTION AND RELATED WORK

Orienting documents is an old problem in tabletop systems and many strategies have been explored [3]. They fall into two categories: explicit orientation methods and automatic orientation methods.

Explicit orientation methods provide widgets or gestures that let users rotate documents. Although simultaneous control of rotation and translation is achievable [5, 11], having to constantly orient and reorient documents can be tedious and distract users from their primary task. Automatic orientation methods free users from the burden of having to rotate documents [8, 10]: each document is adequately oriented depending on its location, allowing users to solely focus on positioning (Figure 1).

One issue with automatic orientation is that it is rarely optimal. For example, circular orientation – i.e., having documents face away from the center of the table – works well for round tables (Figure 1a) but not for rectangular tables (Figure 2). Although orientation methods can be
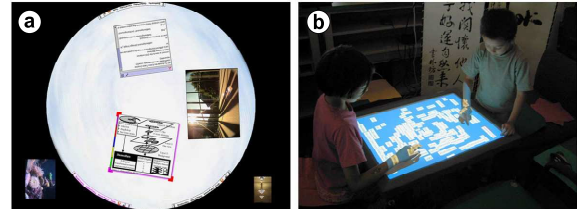
Figure 1. Automatic document orientation in *DiamondSpin*, on a round table and on a square table. Image courtesy of [8].
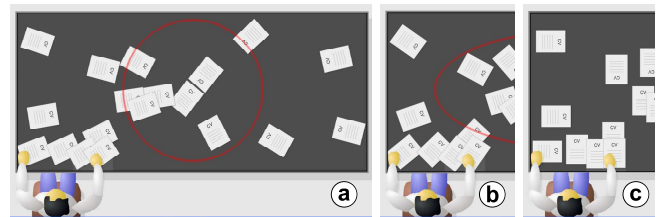

Figure 2. With a circular (a) or elliptic (b) orientation scheme, the user sitting on the corner does not have his documents correctly oriented ; Even with the closest border method (c), two of his documents are still inconveniently oriented.

carefully designed for specific situations, they will always lack the flexibility and the expressiveness provided by explicit rotation, especially when collaborating with other users [4].

To sum up, none of the two strategies proposed so far is satisfactory. Explicit rotation can be tedious and distract users from their primary task. Automatic orientation is convenient, but a method that works well in one case will fail as soon as the context – e.g., the shape of the table, the number and location of users, their tasks or collaborative status – changes. This leads us to propose a middle-ground approach, where users are given control over automatic orientation methods.

This approach first requires automatic orientation methods to be reified into concrete entities. We use vector fields, i.e., functions that map locations to 2-D vectors. Vector fields are a popular mathematical construct, with a well-defined algebra. Besides, they are widely used to describe physical phenomena such as fluid motion or gravitational fields, and hence have well-known visual representations.
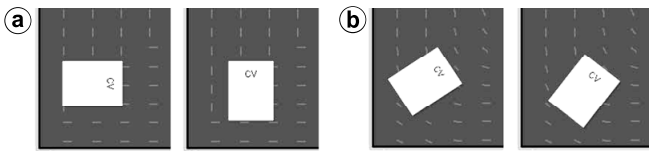
Figure 3. The closest border method, without (a) and with (b) low-pass filtering.

Figure 4. Visualizing elliptic, closest-border, and smoothed closest border orientation fields using vector grids (a) and direction-to-hue (b). *(best viewed in color)*

Since vector fields encode document orientation, we call them *orientation fields*. The design space of the orientation field framework can be split up into three separate problems: a) *generation* of orientation fields; b) *visualization* of orientation fields; and c) *user manipulation* of orientation fields. We examine them in order, before providing a few general illustrations.

GENERATING ORIENTATION FIELDS
Default orientation fields need to be generated before they can be edited. Existing automatic orientation methods (Figure 1) can be described with vector fields, using either continuous functions or a bilinear interpolation of discrete vector fields. On a tabletop system, several such fields can be provided as default configurations. Users would then select them and possibly modify them according to their needs. Modified fields could in turn be saved and retrieved across working sessions.

It is also possible to automatically produce new orientation fields from existing ones. Two potentially useful operations are *filtering* and *combination*.

Filtering: Moving documents can sometimes yield discontinuous changes in orientation (Figure 3a). Since abrupt visual changes are known to confuse users [1], low-pass filtering techniques can be used to eliminate these discontinuous changes. Smooth orientation fields additionally offer more flexibility for orienting documents: the low-pass-filtered closest border method, for example, allows diagonal orientation (Figure 3b).

Combination: New orientation fields can be produced by combining existing orientation fields. Such operations are useful when a consensus needs to be reached, e.g., when several users having their own preferred field happen to sit together around a table, or when several tables are physically assembled [9]. Orientation field combination is also useful for smoothly animating discrete changes in orientation fields, e.g., when browsing orientation fields from a repository.

In practice, orientation fields can be combined by computing a linear combination of vectors in either Cartesian or polar coordinates.

When *Cartesian coordinates* are used, vector length encodes orientation weight or "importance". This allows users to, e.g., define a personal area [7] whose orientation field will not be affected when other users join the table (very long vectors) while leaving the rest open to other users (null-length vectors).
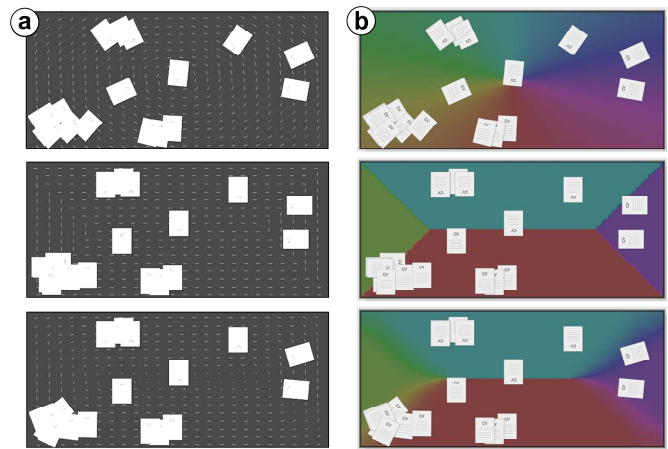
When *polar coordinates* are used, all vector orientations have equal importance. Vector length can be set to an arbitrary constant or used to encode additional information such as document size [10].

VISUALIZING ORIENTATION FIELDS
Since users will be allowed to modify orientation fields, they need visual feedback on the effects of their actions. They also need the behavior of the orientation field to be always visible to be able to use it efficiently. Although this behavior can be suggested by specific visual designs (Figure 1a), we need to be able to visualize arbitrary fields.

Displaying vector fields is a well-researched topic in scientific visualization, with methods ranging from simple vector grids to more sophisticated textures and streamlines [6]. We experimented with two simple ones: *vector grid* and *direction-to-hue*.

Vector grid. This classic technique shows vectors sampled on a regular grid. Rotating vectors by 90° yields a different visual effect (Figures 3 and 4a).

Direction-to-hue. Orientations naturally map to color hues, since hue is cyclic and typically expressed in angular units (Figure 4b). Vector lengths can be either ignored or mapped to other color components such as brightness or saturation.

We found direction-to-hue to be a promising approach. Since hue is preattentive, areas with homogeneous orientations pop out and appear as distinct regions (Figure 4b). As we will further see, this allows to rapidly identify meaningful areas.

MANIPULATING ORIENTATION FIELDS
There are many possible ways to edit orientation fields, and the best methods depend on the hardware available. We explored a technique for the uTable system, which supports direct touch and laser pointing [9]. Our technique lets users specify centripetal orientation fields – i.e., all vectors are

**Figure 5. Reorienting a region, with
k = 1 (the center of attraction is on the table's edge)
and k = 4 (the center of attraction is behind the user).**

oriented towards a point – within regions of arbitrary shape. The technique requires performing two successive gestures, one for specifying the region to be modified, the other one for indicating the center of attraction (Figure 5a).

The system connects the two ends of the first gesture to the nearest edges of the table and closes the curve. The center of attraction C is computed from the two ends $P0$ and $P1$ of the second gesture according to the following formula: $C = k \cdot P0P1$. Choosing $k > 1$ allows C to be off-table, which can be desirable in some cases. For example, personal documents might be more conveniently oriented if they are facing a point behind the observer (Figure 5c) rather than a point on the table (Figure 5b).

ILLUSTRATIONS

We illustrate the use of orientation fields for visualizing and manipulating document orientation with three simple scenarios. These involve the smoothing filter, the direction-to-hue visualization and the centripetal reorientation technique described in the previous sections. We assume a table equipped with a laser pen [9] with which users can drag documents anywhere on the table. The default

orientation field used by the table is a low-pass-filtered closest border method.

Creating a Personal Space

John enters the room and sits at the table next to his colleague Mike (Figure 6a). He starts opening documents but some of them are inconveniently oriented (Figure 6b), so he creates a personal working area oriented towards himself (Figure 6cd). All his documents are immediately reoriented, and all documents he will subsequently drop within that space will be oriented the same way.

If other people join the table, they will see the space John occupies – the red area – even if there is no document on it. This will help them choosing where to sit and where to create their own personal space.
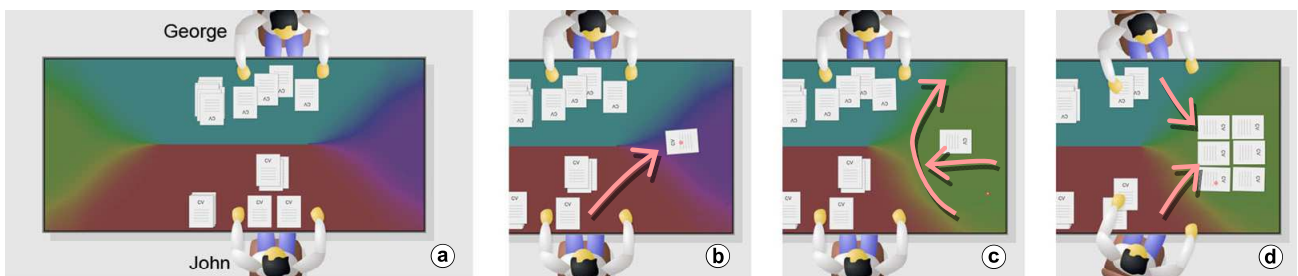
Creating a Shared Space

John and George are inspecting resumes from prospective graduate students and decide to select the top six (Figure 7a). John drags the resume of his favorite candidate to an empty area so that both George and him can see it, but he notices it is upside-down (Figure 7b). With two rapid gestures he re-orients the whole area, turning it into a shared working space (Figure 7c). Orienting the remaining documents is quite straightforward since it only involves dropping them to the green area (Figure 7d).

Temporary Reorientation

Three people are working on an interactive table they reconfigured to suit their needs (Figure 8). John wants to bring Paul's attention to two of his documents. He reorients them with two rapid gestures. Upon reading them, Paul reorients the documents towards John.
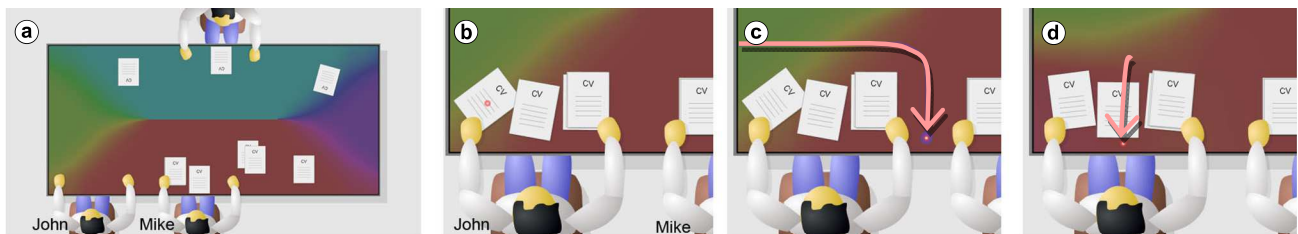
Figure 6. Creating a personal working space.

Figure 7. Creating a shared working space.

Figure 8. Temporarily reorienting documents.

PROTOTYPE IMPLEMENTATION

All the figures except Figure 1 are screenshots of *juTable*, a Java library for creating interactive tabletop mockups running on desktop computers, with user avatars that can be freely arranged around tables of arbitrary size. Such simulations make it much easier to prototype and communicate new interaction ideas. We are currently porting *JuTable* to the *uTable* system [9] so that mockups can be directly deployed in a real setting. The orientation field demo and source code can be downloaded at http://www.aviz.fr/jutable.

CONCLUSION AND FUTURE WORK

We proposed a conceptual framework for managing document orientation, where orientation is controlled by vector fields that can be visualized and edited. This strategy combines the advantages of previous methods: like automatic methods, it maps locations to orientations so that users do not have to repeatedly reorient documents. But this mapping can also be opportunistically modified by users, hence approaching the flexibility of manual rotation.

One noteworthy property of the technique is its visibility: color-to-hue visualization makes areas with homogeneous orientation clearly visible, which turns rotations tasks into simple spatial target acquisition tasks and is likely to promote territoriality and group awareness [7].

The ideas we proposed have been explored and illustrated using mockups, and need to be tested and validated in physically realistic setups. A better understanding on how people use document orientation is also critical to validate the approach [4].

The design space can be further explored in several directions. This includes testing more sophisticated visualizations [6] and exploring physical metaphors such as fluids and forces. For example, users could place magnetic or gravitational objects on the table to modify vector fields, or manipulate conservative vector fields as 3D surfaces. Another open problem is the integration of explicit rotation with automatic rotation techniques.

REFERENCES

1. Chang, B. and Ungar, D. 1993. Animation: from cartoons to the user interface. In *Proc. UIST'93* (Atlanta, Georgia, United States). ACM, New York, NY, pp. 45-55.

2. Forlines, C., Shen, C., Vernier, F. and Wu, M., Under My Finger: Human Factors in Pushing and Rotating Documents Across the Table. in *Proc. INTERACT 2005*: (Rome, Italy, 2005), pp. 994-997.

3. Hancock, M., Carpendale, S., Vernier, F., Wigdor, D., and Shen, C. Rotation and Translation Mechanisms for Tabletop Interaction, tabletop. *First IEEE International Workshop on Horizontal Interactive Human-Computer Systems (TABLETOP '06),* 2006, pp.79-88.

4. Kruger, R., Carpendale, S., Scott, S. D., and Greenberg, S. 2003. How people use orientation on tables: comprehension, coordination and communication. In *Proc. GROUP'03* (Sanibel Island, Florida, USA, Nov. 09 - 12, 2003). ACM, New York, NY, pp. 369-378.

5. Kruger, R., Carpendale, S., Scott, S. D., and Tang, A. Fluid integration of rotation and translation. In *Proc. CHI'05* (Portland, Oregon, USA, April 02 - 07, 2005). ACM, New York, NY, pp. 601-610.

6. Laidlaw, D. H., Kirby, R. M., Jackson, C. D., Davidson, J. S., Miller, T. S., da Silva, M., Warren, W. H., and Tarr, M. J. Comparing 2D Vector Field Visualization Methods: A User Study. *IEEE Transactions on Visualization and Computer Graphics* 11, 1 (Jan. 2005), pp. 59-70.

7. Scott, S. D., Sheelagh, M., Carpendale, T., and Inkpen, K. M. 2004. Territoriality in collaborative tabletop workspaces. In *Proc. CSCW'04* (Chicago, Illinois, USA, Nov. 06 - 10, 2004). ACM, New York, NY, 294-303.

8. Shen, C., Vernier, F. D., Forlines, C., and Ringel, M. DiamondSpin: an extensible toolkit for around-the-table interaction. In *Proc. CHI'04* (Vienna, Austria, April 24 - 29, 2004). ACM, New York, NY, pp. 167-174.

9. Shi, Y., Yu, C., Shi, Y., Finger gesture interaction on large tabletop for sharing digital documents among multiple users. In *First IEEE International Conference on Ubi-Media Computing,* 2008. pp. 8-13.

10. Vernier F., Lesh N. and Shen C. Visualization techniques for circular tabletop interfaces. In *Proc. AVI'02* (Trento, 24-16 May 2002, Italy), pp 257-263.

11. Wu, M. and Balakrishnan, R. Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. In *Proc. UIST'03* (Vancouver, Canada, November 02 - 05, 2003). ACM, New York, NY, pp. 193-202.