# Supporting Display Scalability by Redundant Mapping

Axel Radloff[1], Martin Luboschik[1], Mike Sips[2], and Heidrun Schumann[1]

[1] Institute for Computer Science,
University of Rostock, Germany
[2] Geoinformatics, German Research Center for GeoSciences,
Potsdam, Germany

**Fig. 1.** Applying the proposed Two-Step-Mapping to generate scatterplots showing the same clusters on different display devices in our smart meeting environment.

**Abstract.** Visual analysis sessions are increasingly conducted in multi-display environments. However, presenting a data set simultaneously on heterogenous displays to users is a challenging task. In this paper we propose a two-step mapping strategy to address this problem. The first mapping step applies primary mapping functions to generate the same basic layout for all output devices and adapts the object size based on the display characteristic to guarantee the visibility of all elements. The second mapping step introduces additional visual cues to enhance the effectiveness of the visual encoding for different output devices. To demonstrate the Two-Step-Mapping we apply this concept to scatter plots presenting cluster data.

## 1 Introduction

Smart environments integrate a multitude of interconnected devices to facilitate a pro-active assistance for multi-user scenarios. Those ensembles consist of stationary devices such as desktop devices, projectors, motion trackers, or wall-sized displays, but also aim to integrate personal devices of users such as laptops, PDAs and smart phones. Smart meeting rooms are a typical application scenario of smart environments [1] serving as a basis to communicate information to facilitate discussions and to support decisions.

However, in [2] the challenge of *Display Scalability* has been described dealing with the consistent visual encoding of the same data on different output devices such as smart phones, laptops or large public displays (see figure 1). The problem to be solved here is related to this challenge and in particular aims to avoid the following problems:

– On *small displays*, visual clutter may occur and
– on *large public displays* connectivity information can be lost.

This means, simultaneously presenting the same information on different output devices is a challenging task to be solved in smart environments.

In this paper, we address the problem of *Display Scalability* through extending the classical visual mapping to visual variables (which we call primary mapping) to a redundant encoding of data (which we call secondary mapping).

We will demonstrate this approach using the example of presenting clustered data in scatter plot displays.

The paper is organized as follows: First we briefly reflect the state of the art in Section 2. In Section 3 we introduce the two-step mapping strategy and exemplarily show the application to scatter plots presenting classified data. Section 4 describes a short user-study and Section 5 concludes and gives an outlook on further work.

## 2 Related Work

Visual representations have been adapted in multiple ways to address different data properties (e.g., [3, 4]), different visualization goals (e.g., [5, 6]) and different user capabilities (see [7]). However, the generation of proper visual representations in consideration of heterogeneous multi-displays, as they are found in smart environments, has not been sufficiently examined. Such environments are generally heterogeneous ensembles, that change over time (joining and leaving devices) and facilitate collaborative work (e.g., in [8, 9]). The specifically developed, rare visualization approaches for those ensembles typically combine the individual displays into a large single one. Thus, current research mainly addresses the problems of *sharing* content synchronously from multiple devices on multiple displays and *sharing* the corresponding multiple interactions on the devices (e.g., [9, 10]). Other research projects in the field of multi-desktop environments study the effectiveness of such environments (e.g., [11]) or of single display types (e.g., [12]). However, they do not provide a strategy on how to adapt the same visual representation to different displays.

The adaptation of graphical representations according to given output devices, in particular considering the reduced display size of mobile devices, has been addressed extensively in different fields (e.g., for maps [13], for 3D-models [14], for images [15]). The underlying techniques – scaling, aggregation, interpolation, progression and sampling – are also applied in the field of information visualization (e.g., [16–18]). However, visual clutter remains the main problem. It appears if displaying too much data on a screen with limited space [19]. On the other hand, enlarging the display size may pull the visual objects apart and hence, features such as data density may be misinterpreted [20, 17]. To score the perceivable visual features and goodness of a visualization and thus, to address these problems, different measures have been introduced (see [21] for an overview).

Such measures in combination with appropriate thresholds, are used either to reduce the amount of displayed data (e.g., by sampling [17, 18]), to simplify the visual representation (e.g., the use of binning in [22, 23]) or to determine an appropriate level of progression [24]).

## 3 Two-Step Mapping

The mapping step specifies the visual encoding of data by defining visual abstractions that can be graphically presented. That means visual abstractions represent data through graphical objects specified by their geometry and additional attributes describing the appearance of the objects. The choice of the visual encoding has a huge impact on the effectiveness of conveying information to the user. The mapping step is influenced by different constraints like the characteristics of the data to be encoded, the capabilities of the human visual system, the tasks at hand, but also the characteristics of the given output displays. Because of the diversity and complexity of constraints, many mapping approaches consider one constraint only (typically the data characteristics). In the case of considering output devices, the mapping primarily takes the limited resources of mobile output devices into account, in particular the limited screen size. However, to the best of our knowledge, no mapping strategies published so far specifically addresses the requirements of simultaneously presenting the same data on different displays.

This leads to two problems to be solved. First, in any case the visibility of the elements to be displayed has to be guaranteed depending on the display resolution and eye distance. Second, the capability of a human when working with different display devices may vary significantly, influenced by nothing but the human cognition (see e.g. [12]).

We address these issues by introducing a two-step mapping process. The basic idea is to distinguish between primary and secondary mapping functions. The *primary mapping functions* generate the basic visualization by realizing the typical mapping step generating a specific visual representation. Additionally, the size of visual abstractions is adapted in such a way that visibility on different display devices is guaranteed.

The *secondary mapping functions* define additional visual cues to redundantly encode information and thus, improve the readability of visual representations in multi-display environments. To this end, the combination of primary and secondary mapping functions preserves visibility and important properties of visual abstractions, but also reflects the characteristics of heterogeneous output devices.

### 3.1 Primary Mapping

The primary mapping corresponds to the classical mapping step of the visualization pipeline. It generates the visual encoding of data according to the characteristics of a given visualization technique. Thus, the primary mapping defines the same basic layout for all displays.

The visibility of visual abstractions on different displays has to be ensured. For this purpose, the size of graphics abstractions needs to be adjusted to a well-defined minimal size to guarantee that objects are perceivable by the user.

The adaptation of the object sizes is based on the procedure for eye testing. According to the ISO standard for visual acuity testing [EN ISO 8596:1996-05], an object is distinguishable from the background and other objects and hence, visible (assuming a visual acuity of 1.0), if the object covers at least 1 arc minute of a humans visual angle. Thus, in contrast to [25] we don't determine the covered visual angle of an object. We assume that the object covered at least 1 arc minute ($\alpha = 1$ arc minute) of a humans visual field. Hence, we calculate the required size ($s$) of the object using a given viewing distance ($d$) by the use of the law of sine:

$$s = d * \tan(\alpha) \tag{1}$$

Furthermore, let $r$ be the ppi (pixels per inch) of the specific display and $s$ the object size in inch, then $p$, the number of pixels, covered by the object, can be determined by the following equation:

$$s * r \leq p \tag{2}$$

Because of the discrete nature of the pixel space, $p$ has to be adjusted upwards to the next integer value to guarantee visible objects.

Note, the visual acuity is influenced by the lightness. Higher lightness would increase and lower lightness would decrease the visual acuity. This was first proven by König [26]. However, a light density between 160 and 320 $cd/m^2$ has no significant influence on the visual acuity [27]. Furthermore, the light density of a typical display is in this range and hence, this parameter need not be specifically taken into account.

Using the typical viewing distances of [28], the required minimum object sizes for different display classes can be pre-computed, and stored by a Look-up-Table.

Based on this, the primary mapping step checks the size of the generated visual abstractions and scales the visual output as necessary.

Since all visual abstractions are scaled in the same way, the visual attribute *size* (e.g. as range [min, max]) can still be used to encode data.

### 3.2 Secondary Mapping

The secondary mapping step redundantly encodes data to improve the effectiveness of visual representations on heterogeneous displays. In principle, all visual attributes that have not been used by the primary mapping functions can be applied to specify the secondary mapping step.

However, each visualization technique requires different visual encodings and thus, different subsets of visual variables to define the primary mapping functions. Furthermore, different tasks and output devices require specific encodings. Hence, a general guideline for the design of secondary mapping functions cannot be established. However, the functions have to be found with regard to the applied visualization technique, the task at hand and the given output device.

### 3.3 Example: Adapting Scatter Plot Displays to different output devices

Exemplarily, in this Section, we will show the application of the two-step mapping concept on scatter plots displaying classified data. The task to be supported is detecting clusters simultaneously on large displays with sparse point distribution or on small displays with dense point regions.

***Primary Mapping*** In our example, generating scatter plots, the primary mapping is defined by position (encoding data values) and color (class membership). Thus, we specified two primary mapping functions $p_1$ and $p_2$ to define the basic layout.

$p_1$ maps data values onto positions
$p_2$ maps class memberships to color (hue)

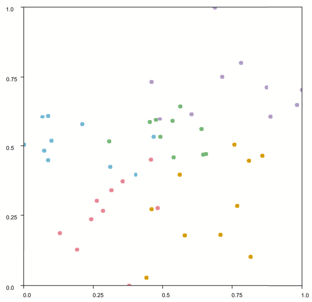
The primary mapping function $p_1$ generates the typical scatter plot display by using *position*, whereas the primary mapping function $p_2$ allows to distinguish the classes, by mapping the cluster membership to *color*.

We encode class membership by *color* because Mackinley [29] ranked this variable to be the second best for nominal data (best is position, which is already used). We apply the color scale provided by Healey et. al. [30] that is proven to be a very good color scale for the distinction of nominal data. However, for gray-scale displays the variable *value* has to be used instead of *hue*.

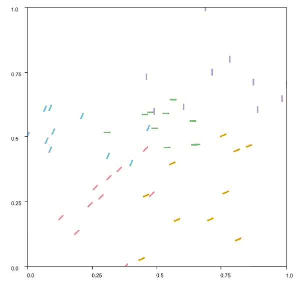
To guarantee visibility of all dots for typical viewing distances we additionally define the mapping function $p_3$:

$p_3$ defines the minimal dot size with regard to visibility constraints.

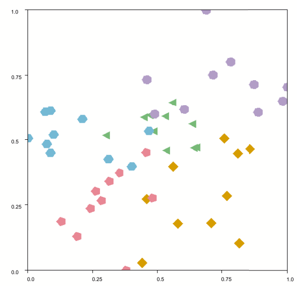
The size calculation is based on the formula described in 3.1.



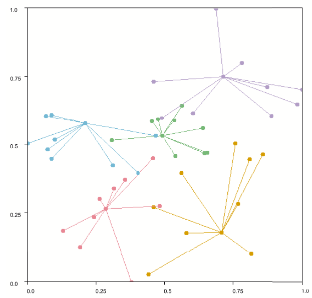
(a) primary mapping



(b) primary + secondary attribute orientation



(c) primary + secondary attribute shape



(d) primary + secondary attribute connection

**Fig. 2.** Scatter plot showing identical data with the application of different mapping functions. (a) shows the classical scatter plot using the primary mapping functions. In (b), (c) and (d) additional secondary mapping functions have been applied; in (b) the secondary mapping function $s_2$ (orientation), in (c) the mapping function $s_3$ (shape) and in (d) the mapping function $s_4$ (connection).

***Secondary Mapping*** For the definition of the secondary mapping functions further appropriate visual variables have to be found to redundantly encode the data, in this specific case the cluster membership. The question is, which visual variables can be used. Our discussion is based on the classification of visual variables by Bertin [31] (position, size, color, value, shape, orientation and texture) and Mackinley [29] who introduced three further attributes: density, connection and containment and replaced *size* by *length* & *area* & *volume* and replaced *orientation* by *angle* & *slope*.

As described above, for redundant encoding of data in the secondary mapping appropriate visual variables have to be found. This procedure depends on characteristics of the chosen visualization technique and the presented data. Using our example of redundant encoding of cluster membership in scatter plots, the following illustrates the individual steps of this procedure:

The first four of the seven visual variables provided by Bertin [31] – position, size, and color (or value for grayscale displays) – are utilized by the primary mapping functions and thus, cannot be used by secondary mapping functions. The visual variable *texture* should also be excluded, since the dot size is adjusted in such a way that the visibility is ensured, but it cannot be assumed that the dot size also ensures the representation of readily identifiable textures. The same holds true for the attribute *containment*. The chosen dot size guarantees visibility, but does not allow for further encodings. Also, a further refinement of visual variables as suggested by Mackinley does not provide new opportunities for our purposes. Readily identifying both (length and area) of small dots would be very difficult. Hence, only the visual variables *shape* and *orientation* of Bertin's list and the variable *connection* of Mackinley's extension can be used as additional variables to adjust the scatter plot display. Based on this discussion, we introduce three additional secondary mapping functions:

1. $s_1$: maps classes onto *orientation*. $s_1$ is realized by replacing the dots representing the data in the scatterplot by little bars. The cluster membership is mapped onto a rotation angle ($\gamma$) with $\gamma \epsilon [0°, 90°]$ and the bars are rotated accordingly. To determine the required angle, the 90° interval is divided equidistantly with regard to the number of existing clusters (see figure 2(b)). Thus, otherwise overplotted dots that do not belong to the same cluster can be distinguished by their differently rotated bars. Here, the minimum size for an object, calculated in the primary mapping, is mapped to the width of a bar.
2. $s_2$: maps classes onto *shape*. For $s_2$ the dots are replaced by regular shapes (all sides have the same length and all angles are the same). Here, the cluster membership is mapped onto the number of vertices. The data value is represented by the center of the shape (see figure 2(c)). The minimum size is mapped to the diameter of the surrounding circle of a shape.
3. $s_3$: maps classes onto *connection*. $s_3$ connects the dots of one cluster with the centroid by lines. Therefore, first the centroid of a cluster has to be found by calculating the center of the cluster $(\frac{x_{max}-x_{min}}{2} + m_{min}, \frac{y_{max}-y_{min}}{2} + y_{min})$ and determining the data point with the least Euclidian distance to the center. Then, the other points are linked to the centroid by drawing a straight line (see figure 2(d)). Here, the minimum size is mapped to the size of the dots.

In the next Section we will describe our user study to demonstrate that secondary mapping functions like $s_1$, $s_2$ and $s_3$ improve the effectiveness of scatter plot displays on heterogeneous displays. The next section shows that this secondary mapping step remarkable improves the perception of clustered data on heterogeneous displays.

## 4 User Study

This user study aims to study how the proposed Two-Step Mapping to present clusters in scatter plots addresses Display Scalability in our smart room environment. To evaluate the Two-Step Mapping strategy, we determine the success rate of connection, shape and orientation for a small, medium and large display device. The success rate is an important issue of Display Scalability and we define it as the ratio of correct answers to all answers.

We compare the success rates of the Two-Step Mapping strategy against the well-known visual mapping of scatter plots which directly corresponds to our primary mapping. In this section, we will present preliminary results of our user study that support our research hypotheses:

1. **H1:** Encodings with secondary mapping perform better on the used display devices in comparison with a single primary mapping of clusters.
2. **H2:** Display device affects the success rate, i.e., the success rate of the different secondary mappings varies across the used small, medium, and large display device.

### 4.1 Setup

**Technical base:** The class of large displays is represented by a 61" TV (see figure 3(b)) with a resolution of 1920$x$1080 pixels (36 ppi (pixels per inch)). The medium displays are represented by a 24" Desktop display with a resolution of 1920$x$1200 pixels (94 ppi) presenting the samples in a 500$x$500 pixel window. The small displays are represented by an Apple iPod touch (see figure 3(a)) 1G with 3.5" and a resolution of 320$x$480 pixels (163 ppi). The viewing distances have been fixed with regards to [28]. Thus, the participants were placed 3m in front of the large TV, 70cm from the desktop monitor and 40cm from the iPod touch.

(a) TV with additional attribute connection

(b) iPod touch with additional attribute orientation

**Fig. 3.** Demonstration of two typical views with applied enhanced mapping.

**Data characteristics:** The dataset to be tested was generated using a mixture of normal distributions: First, the number of clusters (4 to 6) is set for each scatter plot. We experienced that the success rate decreases with the increasing number of clusters for scatter plots on desktop devices; the task to detect clusters in scatter plots becomes difficult for many clusters. Since our aim is to study the relationship between success rate and display device, we restricted the the number of clusters between 4 and 6. The $(x, y)$ coordinates of the centroids of the clusters are drawn from the mixture of normal distributions. Data points are placed around each centroid by considering the centroid as the mean of a normal distribution. To ensure comparability, each view includes a constant number of 200 data points assigned to each cluster.

**Participants:** The user group was composed of 23 non-visualization experts (6 female and 17 male) with an average age of about 34, a minimum age of 18 and a maximum age of 60 years.

**Task:** The task for the participants was to state the number of clusters that can be observed in a scatter plot display. The time for each scatter plot display was restricted to 15 seconds.

**Briefing and Execution:** To brief the participants, test-samples were shown and described in detail. Then, participants were shown scatter plots with 20 different data samples, 5 samples per visual encoding (without additional visual cues, with connection, with orientation and with shape). To avoid a learning effect, the sequence of the display devices, the secondary mappings, and the number of clusters were randomly chosen for each participant.

### 4.2 Results

**H1:** The primary mapping leads to success rates between 63.5% for large displays and 80.0% for medium size displays. The particular low success rate for large displays might be caused by a low density, i.e., the ratio of data points to screen space is low. The secondary mapping clearly improves the success rates in comparison with the primary mapping: for small displays in every case, for medium and large displays in two of three cases in comparison to color (see also 1).

**H2:** Table 1 shows that display size affects the success rate of the different mappings. Whereas, *connection* always outperforms the other mappings, table 1 shows significant differences for the success rate of the secondary mappings. However, the good performance of *connection* was a surprising finding for us since the secondary mapping to connection seems to be related to the corresponding Gestalt law which seems to maintain high success rates on all display sizes.

| mapping | small display | medium display | large display |
|---|---|---|---|
| connection | 89.6% | 87% | 97.4% |
| shape | 87.7% | 73% | 80% |
| orientation | 86.9% | 80.8% | 61.7% |
| only primary mapping | 75.6% | 80% | 63.5% |

**Table 1.** Success rate of the user-study for the different displays and mappings.

## 5 Conclusion and Future work

In this paper we introduced a two-step mapping strategy distinguishing between primary and secondary mapping functions. The primary mapping defines the same basic representation for all displays additionally taking the display characteristics and the typical viewing distances into account. The secondary mapping functions adapt the visual representation in terms of different output devices.

We also applied the two-step-mapping to map based visualizations to redundantly encode data. Here, the graphical elements are area objects, allowing to use another set of additional visual attributes (e.g. texture). Furthermore, we are planing to extend this mapping to further techniques (e.g. Parallel Coordinates and Treemaps).

An interesting topic for future work is the investigation of further visual variables, e.g. provided by NPR techniques. An interesting topic for further investigations could be related to real viewing distances that can in principle be provided by our smart lab.

**Acknowledgments** Axel Radloff is supported by a grant of the German National Research Foundation (DFG), Graduate School 1424 (MuSAMA).

## References

1. Burghardt, C., Reisse, C., Heider, T., Giersich, M., Kirste, T.: Implementing scenarios in a smart learning environment. In: Proceedings of 4th IEEE International Workshop on PervasivE Learning, Hongkong (2008)
2. Cook, K., Thomas, J.: Illuminating the path: The research and development agenda for visual analytics. (2005)
3. Robertson, P.K.: A methodology for choosing data representations. IEEE Computer Graphics and Applications **11** (1991) 56–67
4. Senay, H., Ignatius, E.: A knowledge-based system for visualization design. IEEE Computer Graphics and Applications **14** (1994) 36–47
5. Merino, C.S., Sips, M., Keim, D.A., Panse, C., Spence, R.: Task-at-hand interface for change detection in stock market data. In: Proceedings of the Working Conference on Advanced Visual Interfaces (AVI'06), New York, NY, USA, ACM (2006) 420–427
6. Tominski, C., Fuchs, G., Schumann, H.: Task-driven color coding. In: Proceedings of the International Conference Information Visualisation (IV'08), Washington, DC, USA, IEEE Computer Society (2008) 373–380

7. Kerren, A., Ebert, A., Meyer, J., eds.: Human-Centered Visualization Environments: GI-Dagstuhl Research Seminar. Springer (2007)
8. Encarnação, J.L., Kirste, T.: Ambient intelligence: Towards smart appliance ensembles. In: From Integrated Publication and Information Systems to Virtual Information and Knowledge Environments. (2005) 261–270
9. Pirchheim, C., Waldner, M., Schmalstieg, D.: Deskotheque: Improved spatial awareness in multi-display environments. In: Proceedings of IEEE Virtual Reality Conference (VR'09), IEEE Computer Society (2009) 123–126
10. Forlines, C., Lilien, R.: Adapting a single-user, single-display molecular visualization application for use in a multi-user, multi-display environment. In: Proceedings of the Working Conference on Advanced Visual Interfaces (AVI'08), ACM Press (2008) 367–371
11. Wigdor, D., Jiang, H., Forlines, C., Borkin, M., Shen, C.: Wespace: The design, development and deployment of a walk-up and share multi-surface collaboration system. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'09), ACM Press (2009) 1237–1246
12. Tan, D.S., Gergle, D., Scupelli, P., Pausch, R.: Physically large displays improve performance on spatial tasks. ACM Transactions on Computer-Human Interaction **13** (2006) 71–99
13. Follin, J.M., Bouju, A., FredericBertrand, Boursier, P.: Management of multiresolution data in a mobile spatial information visualization system. In: Proceedings of the International Conference on Web Information Systems Engineering Workshops (WISEW'03), Los Alamitos, CA, USA, IEEE Computer Society (2003) 92–99
14. Huang, J., Bue, B., Pattath, A., Ebert, D.S., Thomas, K.M.: Interactive illustrative rendering on mobile devices. IEEE Computer Graphics and Applications **27** (2007) 48–56
15. Avidan, S., Shamir, A.: Seam carving for content-aware image resizing. ACM Transactions on Graphics (Proceedings of SIGGRAPH'07) **26** (2007) 10
16. Büring, T., Reiterer, H.: Zuiscat: Querying and visualizing information spaces on personal digital assistants. In: Proceedings of the International Conference on Human Computer Interaction with Mobile Devices & Services (MobileHCI'05), ACM Press (2005) 129–136
17. Bertini, E., Santucci, G.: Improving 2d scatterplots effectiveness through sampling, displacement, and user perception. In: Proceedings of the International Conference Information Visualisation (IV'05), Washington, DC, USA, IEEE Computer Society (2005) 826–834
18. Ellis, G., Dix, A.: Enabling automatic clutter reduction in parallel coordinate plots. IEEE Transactions on Visualization and Computer Graphics (Proceedings of InfoVis'06) **12** (2006) 717–724
19. Ellis, G., Dix, A.: A taxonomy of clutter reduction for information visualisation. IEEE Transactions on Visualization and Computer Graphics (Proceedings of InfoVis'07) **13** (2007) 1216–1223
20. Bertini, E., Santucci, G.: Quality metrics for 2d scatterplot graphics: Automatically reducing visual clutter. In: Proceedings of the International Symposium on Smart Graphics (SG'04), Springer (2004) 77–89
21. Bertini, E., Tatu, A., Keim, D.: Quality metrics in high-dimensional data visualization: An overview and systematization. In: InfoVis 2011 – IEEE Information Visualization Conference 2011. (2011, Providence, USA)

22. Fuchs, G., Thiede, C., Sips, M., Schumann, H.: Device-based adaptation of visualizations in smart environments. In: Workshop Collaborative Visualization on Interactive Surfaces (CoVIS), IEEE VisWeek 2009. (2009)
23. Novotny, M., Hauser, H.: Outlier-preserving focus+context visualization in parallel coordinates. IEEE Transactions on Visualization and Computer Graphics (Proceedings of Vis'06) **12** (2006) 893–900
24. Thiede, C., Schumann, H., Rosenbaum, R.: On-the-fly device adaptation using progressive contents. In: Intelligent Interactive Assistence and Multimedia Computing, Proceedings of IMC'2009, Springer Berlin Heidelberg (2009) 49–60
25. Ware, C.: Information Visualization: Perception for Design. Morgan Kaufmann (2000)
26. König, A.: Die abhängigkeit der sehschärfe von der beleuchtungsintensität. In: Sitzungsbericht der Königlich Preussischen Akademie der Wissenschaften zu Berlin 26 (1897) pp. 559 – 575. (1897)
27. Kaufmann, H.: Strabismus. Georg Thieme Verlag (2003)
28. Terrenghi, L., Quigley, A., Dix, A.: A taxonomy for and analysis of multi-person-display ecosystems. Personal and Ubiquitous Computing **13** (2009) 583–598
29. Mackinlay, J.: Automating the design of graphical presentations of relational information. ACM Transactions on Graphics **5** (1986) 110–141
30. Healey, C.G.: Choosing effective colours for data visualization. In: Proceedings of IEEE Visualization (Vis'96). (1996) 263–270
31. Bertin, J.: Graphics and Graphic Information-Processing. de Gruyter (1981)