Visual and Analytical Extensions for the Table Lens

Mathias John, Christian Tominski, Heidrun Schumann

Visual and Analytical Extensions for the Table Lens

Mathias John[a], Christian Tominski[a], and Heidrun Schumann[a]

[a]Institute for Computer Science, University of Rostock,
Albert-Einstein-Straße 21, 18059 Rostock, Germany

ABSTRACT

Many visualization approaches teach us that ease of use is the key to effective visual data analysis. The Table Lens is an excellent example of a simple, yet expressive visual method that can help in analyzing even larger volumes of data.

In this work, we present two extensions of the original Table Lens approach. In particular, we extend the Table Lens by Two-Tone Pseudo Coloring (TTPC) and a hybrid clustering. By integrating TTPC into the Table Lens, we obtain visual representations that can communicate larger volumes of data while still maintaining precision. Secondly, we propose to integrate a data analysis step that implements a hybrid clustering based on self-organizing maps and hierarchical clustering. The analysis step helps to extract and communicate complementary structural information about the data and also serves to drive interactive information drill-down.

Keywords: Information Visualization, Two-Tone Pseudo Coloring, Table Lens

1. INTRODUCTION

Today's datasets are growing in size and complexity. The challenge of visualizing larger datasets is more current than ever before. In order to extract global information from such data efficiently, a compact representation that provides overview and structural insight is needed. On the other hand, it is important to display data values very precisely, because otherwise detail information gets lost. Combining overview and detail is not always an easy task – for both visualization designers as well as users.[1]

We think that simplicity is a requirement for user acceptance and a wider range of applicability. In this work, we adhere to simple, yet expressive visualization methods. In particular, we present two extensions of the well-known Table Lens,[2] which is an elegant approach potentially useful in many applications domains. The first extension concerns the integration of Two-Tone Pseudo Coloring (TTPC)[3] into the Table Lens. While the Table Lens is particularly suited to provide overview and detail for relational datasets with a large number of data items, TTPC has been developed to provide compact overviews of larger value ranges. At the same time, TTPC increases the precision that can be achieved when reading details from a visualization. Consequently, the goal is to combine the strengths of Table Lens and Two-Tone Pseudo Coloring. The second extension concerns the integration of analytical methods into the Table Lens. With the help of an additional analysis step, we are able to rearrange table rows according to clustering information. Our hybrid clustering approach can be seen as a complementary feature to the sorting functionality of the original Table Lens to gain structural insight into the data being visualized.

In Section 2, we provide brief background information to lay the ground for extending the Table Lens by TTPC in Section 3. There, we will also describe a heuristic to improve readability of TTPC-based visualization significantly. In Section 4, we will explain how to integrate self-organizing maps (SOM) and hierarchical clustering in order to generate visual representations that convey structural information as well. Section 5 is devoted to further extensions and visual examples of the proposed approaches. A summary and an outlook on future work will be given in Section 6.

Further author information: (Send correspondence to Mathias John)
Mathias John: E-mail: mathias.john@informatik.uni-rostock.de, Telephone: +49 381 498 7441
Christian Tominski: E-mail: ct@informatik.uni-rostock.de, Telephone:+49 381 498 7494
Heidrun Schumann: E-mail: schumann@informatik.uni-rostock.de, Telephone: +49 381 498 7490

2. BACKGROUND

To facilitate understanding of later sections, we will provide a brief view on the methods we use in this work. We will take a look at the role of color and consider the importance of overview+detail and precision in visualization.

2.1 Two-Tone Pseudo Coloring

In his *Semiology of Graphics* Jacques Bertin describes color as an important visual variable. However, proper use of color is a requirement to fully exploit the potential of color coding.[4] Still, many visual representations make use of rather basic color coding schemes, even though better approaches are available in the literature.[5,6] Presumed such methods are used, color coding is an effective means to visualize qualitative and quantitative data. However, it is not always easy for users to read data values precisely when looking at color coded visual representations. Fig. 1 shows examples of discrete and continuous color coding. In the former case, only ranges of colors, and hence, only ranges of data values can be identified. In contrast to that, continuous color coding is based on a scale of smooth color transitions. However, users tend to perceive a sequence of colors as a set of discrete ones.[7] So, even with very smooth transitions, values cannot be read very accurately.

Recently, Two-Tone Pseudo Coloring (TTPC) has been introduced as a technique to facilitate visualization of univariate quantitative data.[3] TTPC uses a discrete color scale that consists of only few colors (less than eight). Whereas classic color coding (irrespective of whether it is discrete or continuous) maps each data value to one color, TTPC uses two colors, which are adjacent in the color scale. These two colors are interpreted in two steps (see Fig. 1). First, the two colors guide users to a particular interval in the value range. If users find that interval interesting, they go into detail: The proportion of use of both colors encodes the precise data value. This two-step interpretation is the basis for overview+detail. While the first step relies on the perception of color, which facilitates overview, the second step is based on the human capabilities to judge lengths, which provides for sufficient detail. Compared to other approaches, TTPC generates very compact, yet precise visual representations. As seen in Fig. 1 line charts require much more screen space than a TTPC visualization.

However, in its original form, TTPC is suitable for univariate data only. To visualize multivariate data, other methods must be applied, as we will see in the next section.

2.2 Table Lens

Multivariate data (i.e., data with more than one variable) are usually modeled as a data table, where columns accommodate variables and rows represent data items. Even though common spreadsheet representations are very intuitive, they cannot be used to display datasets with a larger number of data items at a glance. The Table Lens[2] addresses this problem. It reduces the required amount of display space by shrinking the height of table rows: Thin bars are used to encode data values. The length or position of a bar serves as indication for the particular data value. To facilitate precise reading of data values, user can expand rows to see data values in textual form. In this sense, the Table Lens supports overview (representation as thin bars) and provides details on demand (value print out).

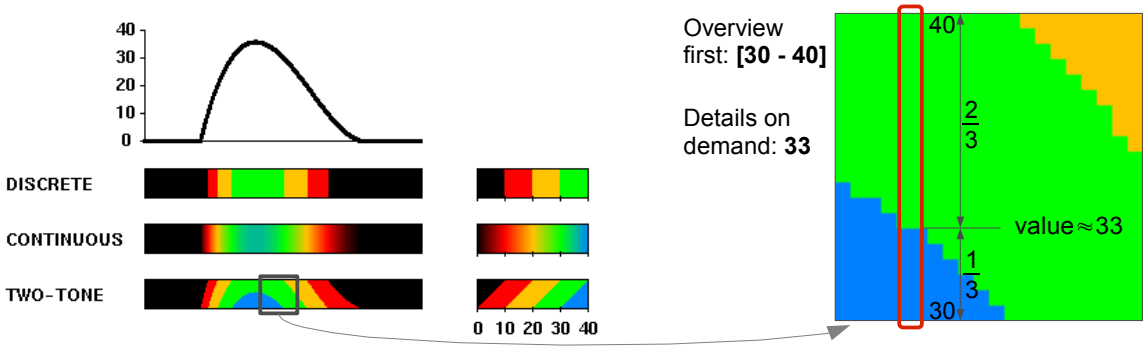

Figure 1. Two-Tone Pseudo Coloring.

In the original Table Lens, overview and detail concern data items (i.e., table rows). In the next section, we will extend the Table Lens by TTPC to achieve more compact representations with respect to variables (i.e., table columns). TTPC encoding results in representations that are based on both perception of color *and* length. With such a two-channel encoding it is easy to maintain an overview, even when table columns are narrow in the case that many attributes are being displayed.

In order to assist users in finding hidden relationships, it makes sense to support rearrangement of table rows. Users of the original Table Lens can sort the data on demand on a column-by-column basis. As we will see in Section 4, providing a hybrid clustering mechanism complementary to standard sorting functionality is helpful to reveal structural information.

All in all, we think that integrating novel visual representation and analytical methods into the Table Lens bears much potential in terms of increasing its usefulness and applicability. With these thoughts, we are in line with other researchers in the field.[8]

3. TWO-TONE PSEUDO COLORED TABLE LENS

In what follows, we introduce a way to join the advantages of Table Lens and TTPC. The basic idea is to replace the original Table Lens bars by TTPC value representations. The result is a Table Lens that shows TTPC views stacked as table columns. The integration of TTPC allows for more columns to be displayed, because TTPC supports precise reading of data values, even if the table cells are narrow.[3] We understand TTPC as a complementary technique for the Table Lens. All features of the original Table Lens are still supported: The intuitive overview+detail concept allows users to change the width of columns and to adjust the height of rows (geometric zoom), and it is possible to expand table rows instantly to see print outs of the data in textual form.

Nonetheless, integrating TTPC into the Table Lens is not without problems. First of all, we have to support users in learning the new way of representing data as TTPC-coded columns. Second, since TTPC involves color, we have to take care to use color scales that integrate smoothly into the data exploration process. In the next paragraphs, we will address these problems.

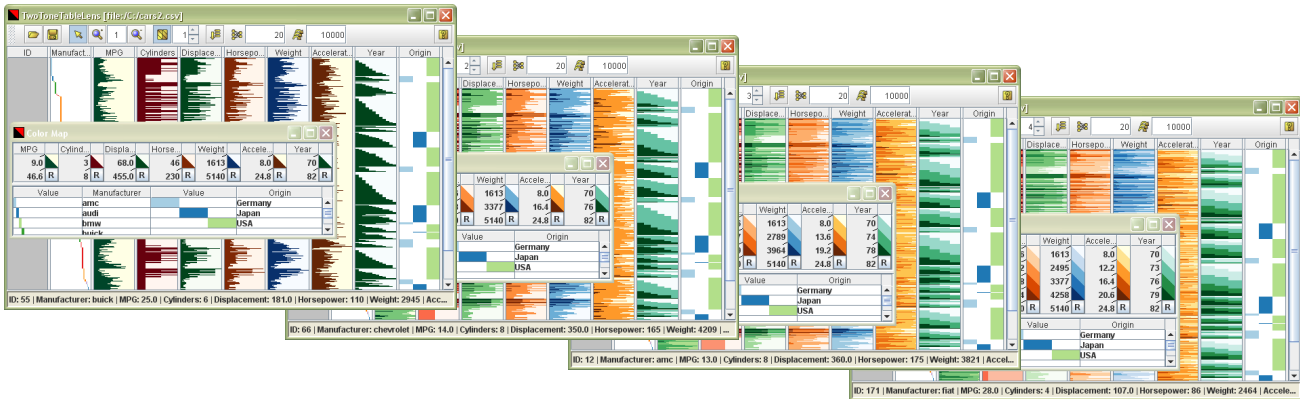

Figure 2. TTPC Table Lens with different numbers of color segments (left: $n = 1$ to right: $n = 4$).

3.1 Learning the TTPC Table Lens

To support the process of learning TTPC Table Lens representations we build a bridge between the classic Table Lens and our approach This is accomplished by so to say interpolating between both techniques. Interestingly, this is possible, because the original Table Lens is a special case of a TTPC Table Lens where n, the number of color segments, equals one. We allow users to set n to any value between one and eight (see Fig. 2). Precision is increased for high values of n, but then users need some experience to interpret the TTPC Table Lens. By setting n to low values, novel users get easy access to a TTPC Table Lens and can get accustomed to the new presentation technique.

3.2 Effective Use of Color

The choice of colors used as well as the segmentation of the color scale have major impact on the effectiveness of the TTPC Table Lens. Consider a color scale with four segments like $black : [0.3-0.99], red : [0.99-1.95], yellow : [1.95-2.91], green : [2.91-3.87]$. Users that are not familiar with wavelengths of colors will hardly understand the order in that color scale. Furthermore, switching colors at odd values impedes reading intermediate values from a TTPC representation. Which colors are better suited and how segmentations like the previous one can be automatically transformed into easier to understand ones will be shown next.

Color scales To visualize quantitative variables with color, it makes sense to map data values to the saturation component.[4] When visualizing multiple quantitative variables it is furthermore very helpful to assign separate color scales to each variable to communicate that each variable has its own value range. Variables are then easier to recognize, which is particularly useful in tabular representations that allow for column reordering. Because we want to use saturation to visualize data values, our only choice to generate separate color scales for the variables is to vary the hue component. We carefully chose eight color scales provided by the ColorBrewer[9] and applied those in a way such that adjacent columns of the Table Lens can be distinguished effortlessly. However, if the TTPC Table Lens is used in specific application scenarios, user preferences or application background (e.g., context-specific color associations or per se correlated variables) may raise the need to use color scales tailored to the requirements at hand.

Color scale segmentation As already indicated, carelessly set color segment borders are a problem. To generate more intuitive color scales, we propose a novel color scale segmentation heuristic. The basic idea is to spread the value range mapped to a color scale such that the subdivision into color segments results in transition points that are easier to interpret. How the heuristic method works can be best explained by our introducing example, where the value range was $[0.3-3.87]$. Assumed the number of color segments is $n = 4$, then a simplistic subdivision of the color scale would result in the following awkward transition points (i.e., segment borders): $p_0 = 0.3, p_1 = 0.99, p_2 = 1.95, p_3 = 2.91, p_4 = 3.87$. In contrast to that, our heuristic method does the following:

1. Find smallest k such that $10 \leq r * 10^k < 100$, with $r = max - min$. Scale value range by setting $max' = max * 10^k$ and $min' = min * 10^k \Rightarrow (max' = 38.7,\ min' = 3)$.
2. Determine smallest i with $(max' + i)$ **mod** $10 = 0$. Set $max'' = max' + i \Rightarrow (max'' = 40)$.
3. Calculate expanded value range $r' = max'' - min'$ and determine distance between transition points $d = r'/n \Rightarrow (r' = 37,\ d = 9.25)$.
4. Determine smallest j with $(d + j)$ **mod** $2 = 0$. Set $min'' = max'' - n * (d + j) \Rightarrow (min'' = 0)$.
5. Scale back to original magnitude **min**$= min'' * 10^{-k}$ and **max**$= max'' * 10^{-k} \Rightarrow$ (**max**$= 4$, **min**$= 0$).

In the first step, the original value range is scaled to a fixed range of 10 to 100. This step is necessary, because our heuristic has to cope with arbitrary magnitudes. In the second step, we set the maximum to a more intuitive value by increasing the scaled maximum to the next multiple of 10. The third step computes the distance between transition points. In step four, we consider the fact that people commonly tend to cut things into halves when they have to make estimations. To support this habit, we increase the computed distance to the next multiple of 2 and determine a new minimum value. Finally, the fifth step is to bring the values back to their original magnitude. Applied to the previously mentioned example, our heuristic spreads the value range such that the transition points read: $p_0 = 0, p_1 = 1, p_2 = 2, p_3 = 3, p_4 = 4$, which is much easier to interpret compared to straightforward color segmentation.

We tested our method with various quantitative variables and recognized that it usually helps to increase the readability of the TTPC Table Lens. Even though we believe that our approach can also be applied to other color-based visualization techniques, a heuristic will never be able to foresee all possible requirements and constraints that may be implied by certain application scenarios. Therefore, we allow users to adjust the

color scale segmentation interactively. Fig. 3 shows a comparison of a simplistic segmentation, the result of our heuristic method, and a segmentation adjusted by a user. It is quite obvious, that the heuristic has advantages over the simplistic method. Whether the heuristic can compete with user-defined segmentations depends on the application context and tasks at hand.

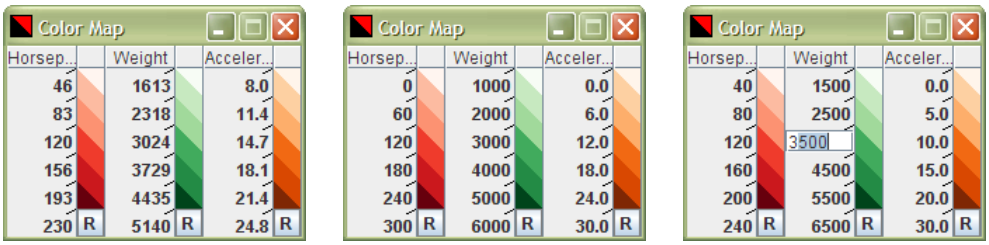

Figure 3. Color scale segmentation – (a) Simplistic segmentations are difficult to interpret; (b) Our heuristic helps in generating more intuitive segmentations; (c) Users can always intervene and adjust the color scale to their particular needs.

4. HYBRID DATA CLUSTERING

So far, we have seen that improved color scales are very useful in terms of readability. A further aspect must be taken into account to make our approach applicable for arbitrary datasets: Stronger oscillation in the data hinder users in spotting relations between variables, and thus degrades efficiency. This is due to the larger number of color switches present in the columns of the tabular representation (see Fig. 4(a)). A color switch is considered if vertically adjacent table cells do not share common colors. In such cases, the cognitive effort is higher, because each cell must be interpreted separately. On the other hand, if adjacent cells show the same colors, the overview (step one in interpreting TTPC) is the same for these cells, only their precise values vary (detail). Users can derive data values more easily. We call this cell-to-cell coherence.

To utilize cell-to-cell coherence, that is to reduce the number of color switches, a mechanism is needed that assigns similar data items (in terms of Euclidean distance) to rows that are close in the visual representation. A first possibility, which is also provided in the original Table Lens, is to sort data values. Sorting data successively with respect to user-chosen columns opens up the chance to arrange similar table rows close together. However, chances decrease with a growing number of variables. Moreover, this way of sorting is time consuming in terms of required interaction steps.

Clustering methods for multivariate data are a promising alternative. The result of clustering is a set of clusters, which contain similar data items. Clustering yields two advantages: First, knowing clusters and their associated data items makes it easy for us to rearrange table rows to avoid strong data oscillation, and second, clustering methods crystallize structure within the data.

Because our interests regard larger datasets, we have to choose a clustering method that generates results within reasonable time. Self-organizing maps (SOM) are well-suited for finding clusters in multivariate data.[10] Yet, their computational complexity does not fully fit our needs. Therefore, we opted for implementing a hybrid clustering that integrates SOM clustering and hierarchical clustering. Our hybrid clustering performs the following two steps:

1. Coarse SOM clustering to generate a reasonable number of SOM clusters,
2. Refine each of the SOM clusters with hierarchical clustering.

We will take a closer look at our hybrid clustering method and provide some test results in the following paragraphs.

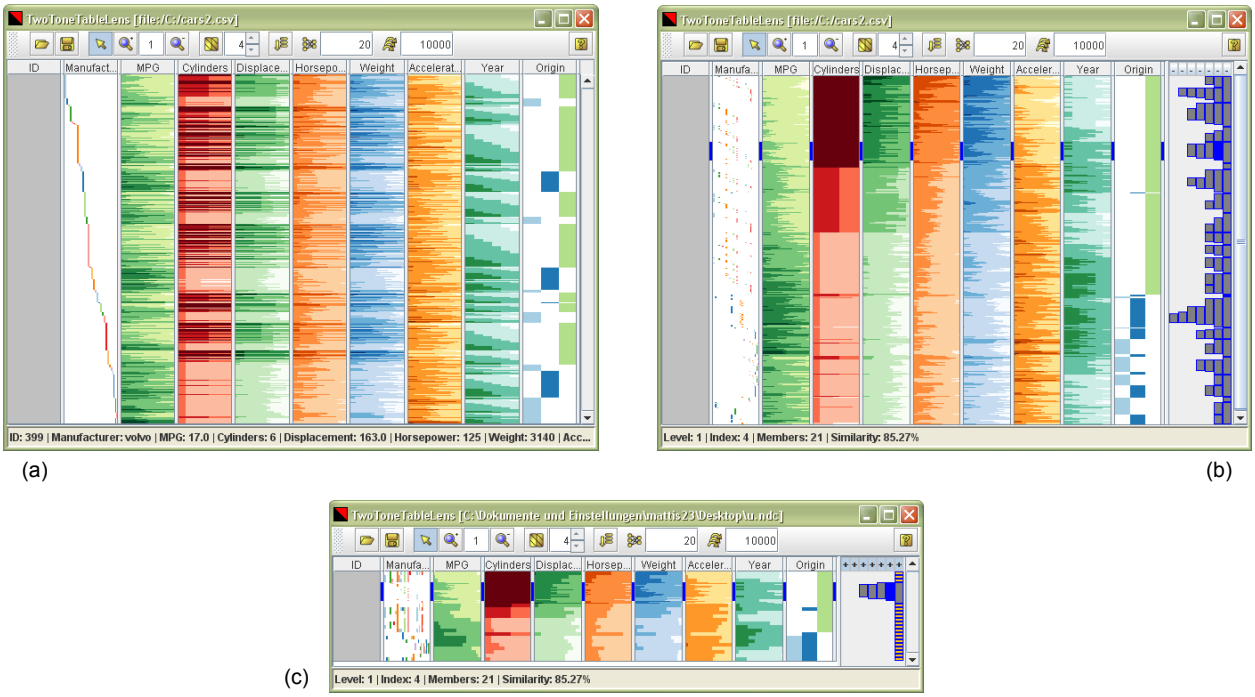

Figure 4. Hybrid clustering to reduce strong data oscillation – (a) Original arrangement of rows with many color switches; (b) Hybrid clustering reduced the number of color switches, the cluster hierarchy is represented in an Icicle Plot (fully expanded); (c) Starting from a fully collapsed Icicle Plot, users can perform information drill-down.

Step one – SOM clustering Self-organizing maps[11] are artificial neuronal networks whose neurons are typically arranged on a rectangular lattice. For our purposes we use a $1D$ variant of SOM, for which the lattice is reduced to a chain. This enables us to use the clustering result directly to rearrange rows in the Table Lens. In a SOM, each neuron is associated with an m-dimensional neuron vector, where m is the number of variables of the dataset. After a network training with randomly chosen training vectors, each neuron vector can be considered the representative of a cluster of similar data vectors. Moreover, adjacent cluster representatives (in terms of their position in the chain of neurons) show similarities. Hence, we have achieved our goal to bring similar data vectors close to each other.

Although SOM are straightforward to implement, the problem is to estimate the number of neurons (i.e., clusters) required to generate results that are balanced in terms of clustering quality and computation time. The proper number of neurons is not known in the first place and depends heavily on the given data. Therefore, we take a closer look at the influence of the number of neurons used. Notice that once the number of neurons is known, the number of training iterations can be determined by the following rule of thumb:[11] $\textit{training iterations} \approx 500 \cdot \textit{number of neurons}$.

Besides extracting structural information, for us the primary goal of clustering is to reduce data oscillation to improve the visual representation. So, roughly speaking, clustering quality can be related to the number of color switches. The occurrence of a single color switch within a certain area is not really a problem, yet, the sequence of many is. Therefore we introduce the term *crucial color switch*. It describes a color switch whose distance to another switch located above in the spreadsheet is less than a certain number of rows. The maximum distance for a crucial color switch is approximately five rows. Color switches with larger distances do not seem to interfere with value reading. Yet, perceptual tests may yield a slightly different threshold. To evaluate clustering results, we created a metric that counts all crucial color switches and relates them to the total number of color switches. For a dataset with m variables and k data vectors, we compute our metric $M(c)$ for each column c of the TTPC Table Lens as follows:

$$M(c) = \frac{\sum_{i=1}^{k} switch(i,c) \cdot crucial(i,c)}{n-1} \tag{1}$$

$$switch(i,c) = \begin{cases} 1 & | \quad S(v_{i-1,c}) \neq S(v_{i,c}) \\ 0 & | \quad \text{else} \end{cases} \tag{2}$$

$$crucial(i,c) = \begin{cases} 1 & | \quad \exists j : i < j \leq i + d \ \wedge \ S(v_{j,c}) \neq S(v_{i,c}) \\ 0 & | \quad \text{else} \end{cases} \tag{3}$$

$$d \approx 5 \tag{4}$$

where $v_{i,c}$ denotes the data value in row i and column c, and $S(v_{i,c})$ is its associated color segment. To get a single value for clustering quality, we compute the average of $M(c)$ of all m columns.

We tested pure SOM clustering quality for different numbers of neurons on datasets of different sizes (see Tab. 1). Since SOM is a nondeterministic method, we conducted the evaluation five times and averaged the final result. Tab. 2 shows that reducing the occurrences of crucial color switches means to significantly increase the number of neurons. That is a problem, because the training of large SOM networks (e.g., 1000 neurons) takes an unreasonable amount of time. This is where the second step of our hybrid clustering comes into the picture.

	climate	*health*	*countries*
variables	10	10	10
data items	40419	1096	110

Table 1. Test datasets and their sizes

Step two – Hierarchical clustering Hierarchical methods introduce a step by step procedure to cluster data vectors.[12] The result is a hierarchical structure with several cluster levels, where the root of the hierarchy is a representative for the whole dataset and the leaves are the original data vectors. The hierarchical structure is very useful from a visualization perspective, because it communicates the process of generating clusters very well.

Hierarchical clustering methods usually require a dissimilarity matrix, which describes the distances between any two clusters. This matrix can become huge and its computation may be too expensive if we consider larger datasets for visualization. Therefore, we do not apply hierarchical clustering to the entire dataset, but only to the generated SOM clusters. We tested this hybrid approach and derived the following interesting conclusion from Tab. 3. Since hierarchical clustering refines SOM clusters, the number of neurons used for the SOM step is rather independent from clustering quality. This independence allows us to reduce the number of neurons without compromising quality and, thus, to accelerate the hybrid clustering. We propose the following rationale: *number of neurons* $= \lfloor \sqrt{k} \rfloor$, where k is the number of data vectors. The tests we conducted indicate that the costs of SOM training and hierarchical clustering are balanced in that case. This has basically two advantages.

neurons	*climate*	*health*	*countries*
0	27.41	23.57	51.20
5	26.52	12.86	30.69
10	25.74	12.71	25.87
20	25.08	12.39	20.03
50	23.50	11.42	–
100	21.63	10.98	–
500	17.19	–	–
1000	15.54	–	–

Table 2. Pure SOM clustering quality – Lower values indicate a lower percentage of crucial color switches, and hence, higher quality; "–" denote test series that were not expected to lead to significant result variations.

neurons	*climate*	*health*	*countries*
5	–	9.70	19.38
10	9.18	9.89	18.61
20	9.29	9.95	16.17
100	9.54	9.77	–
200	9.62	9.49	–

Table 3. Hybrid clustering quality – Lower values indicate a lower percentage of crucial color switches, and hence, higher quality; "–" denote test series that were not expected to lead to significant result variations.

First, we can reduce the number of neurons required for the SOM step. Second, we avoid huge dissimilarity matrices, since the SOM clusters to be refined contain only a limited number of data vectors.

We can conclude that by integrating our hybrid clustering into the Table Lens strong oscillation in the data can be reduced significantly. Fig. 4(b) shows the visual effect: The overview is much easier to grasp and precise values can be read with less effort. Moreover, hidden multivariate relationships within the data can be revealed, which is an advantage over plain sorting functionality. It is worth mentioning that even for larger datasets, the hybrid approach is still capable of reflecting the underlying data characteristics very well, without putting too much burden neither in terms of computation nor in terms of memory usage.

5. FURTHER EXTENSIONS & EXAMPLES

5.1 Further Extensions

In order to widen the range of applicability of the TTPC Table Lens, we considered two further aspects:

- Communication of underlying data characteristics
- Reproducibility of visualization results

The hybrid clustering process induces a hierarchical structure that is very well suited to communicate underlying data characteristics. We make use of that fact by providing an additional Icicle plot to visualize the cluster hierarchy. An Icicle Plot requires less space than other designs[13] and can be integrated easily by inserting an additional column into the Table Lens (see Fig. 4(b)). Table Lens and Icicle Plot are linked, so that we achieve a very tight coupling of data representation and visualization of underlying structure. We also integrated homogeneity information derived from the hierarchical clustering step. This information is valuable when grading similarity between cluster elements. Lastly, the Icicle Plot drives a semantic zoom facility: Each element in the Icicle Plot and its header row can be clicked to expand single clusters or entire levels of the cluster hierarchy respectively. Thus, starting with a completely collapsed Icicle Plot, users can interactively drill-down to areas of interest; the Table Lens is updated accordingly (see Fig. 4(c)).

A second extension regards reproducibility. We think that being able to reproduce visualization results is a key to success of any visual method. Since our approach involves a nondeterministic analysis step (SOM clustering), reproducing a particular result can be an awful task. Therefore, we incorporated a mechanism that enables users to save the current state of the visualization, including color scale, clustering results, current state of the Icicle Plot. That allows users to come back later, either to revisit the visualization result or to continue with their analysis task. The saving functionality is also useful when comparing visualization results that were generated using different parameterizations.

5.2 Examples

In the previous sections, several extensions to the Table Lens approach were introduced. The potential of the extended Table Lens evolves from its different features, including overview+detail concepts, functionality to increase readability (i.e., conscious use of color scales and color segmentation), integration of a hybrid clustering linked to an interactive drill-down facility, and mechanisms to support reproducibility. We implemented our approach in a tool[14] to get an impression of how well the features can be applied to certain problems in practice. In what follows, three different examples will be explained.

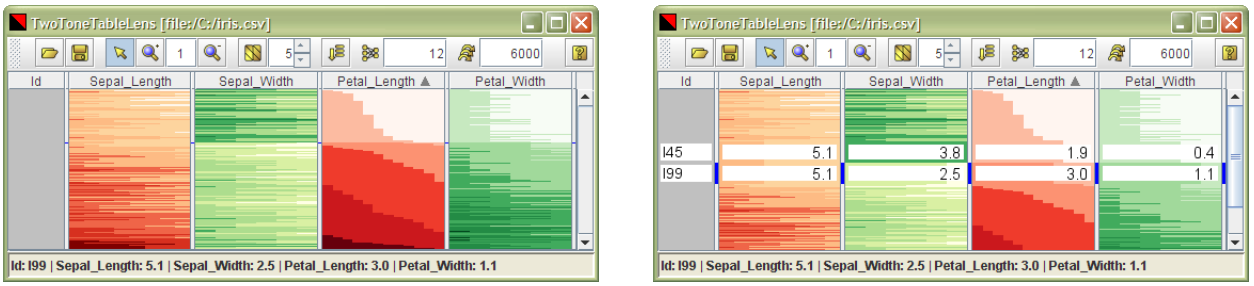

Figure 5. Iris dataset – (a) Data sorted with respect to petal length; (b) Rows adjacent to discontinuity are expanded to get detail information.

Iris dataset The well-known Iris dataset[15] describes Iris blossoms. Since it contains only 4 variables (plus an additional identifier) and only 150 data vectors, we omit the clustering steps and go directly into visual detail. Let us sort the data with respect to petal length (4th column). At first view a discontinuity in the middle of that column can be recognized (see Fig. 5(a)). The area describes a value step which is silhouetted against a smooth value development. Supported by TTPC overview and an appropriate color scale this area can be found immediately. Using Table Lens' mechanism to expand rows of interest, we can easily confirm a numeric gap (see Fig. 5(b)), meaning that the dataset holds no Iris exemplar with a petal length between 1.9 cm and 3.0 cm.

Countries dataset "Countries" is a dataset describing the census of 110 countries in the early 90s. As can be seen in Fig. 6(a) the representation of the original dataset looks rather abstruse. Therefore, we applied hybrid clustering using ten neurons as suggested by our tool. The resulting visualization looks smoother and is easier to interpret. By taking a closer look, one can find an interesting cluster that contains exclusively former east block countries (see Fig. 6(b)). The Table Lens view clearly shows, that only these countries share an exclusive property, that is the combination of high literacy rate with low GDP, low birth rate and rather high death rate (see Fig. 6(c) columns 4,6, 7, and 8).

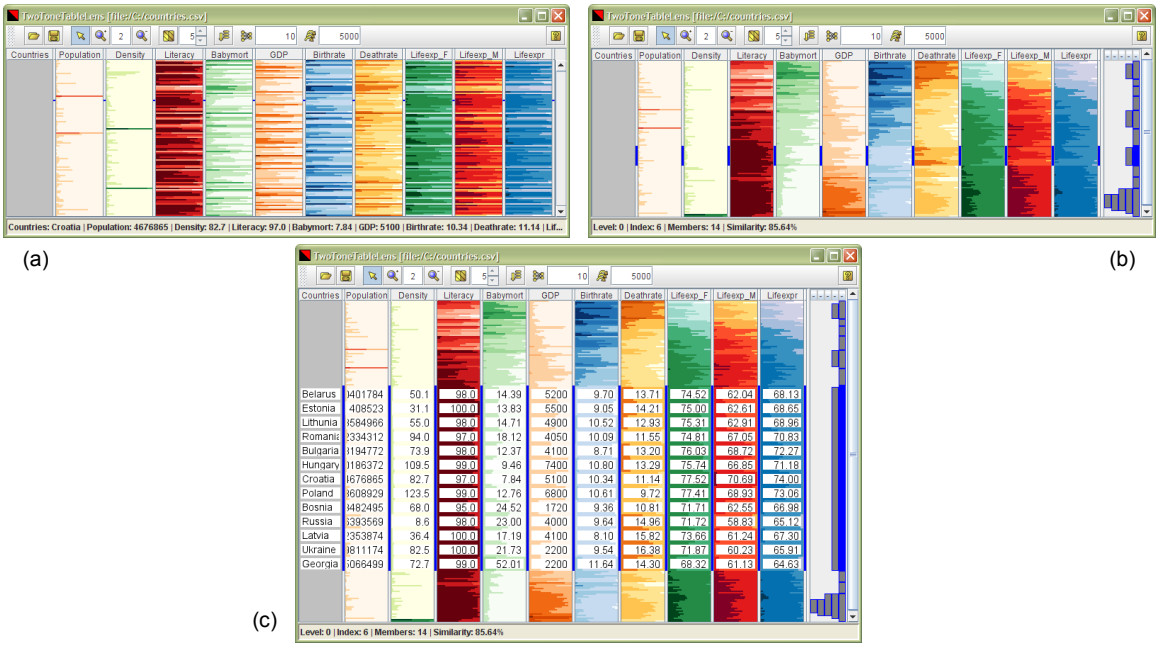

Figure 6. Countries dataset – (a) Unsorted view; (b) After clustering has been applied, a cluster of interest can be selected (denoted by blue bar); (c) Expanding that cluster reveals former east block countries.

Climate dataset We now consider a larger dataset related to climate research.[16] It contains 40419 data vectors about daily weather conditions, including for instance temperature, air pressure, precipitation, sunshine duration etc., at the Potsdam observatory. By looking at Fig. 7(a) one can see a smooth value development in most of the columns, which is due to the fact that weather usually changes gradually. From looking at just the first 365 data vectors, we may guess that a correlation exists between temperature and vapor content (4th and 5th column). However, since only a fraction of the data is visible, we cannot be confident. What we could do in such a case is to scroll all the way through the data to confirm our guess. Yet, that would be rather inconvenient in terms of navigation, since we are dealing with a rather large dataset. The alternative is to apply clustering to the data first, and then to collapse all the elements of the Icicle Plot by a single click. This results in an overview that shows average values for each cluster (see Fig. 7(b)). This overview is much easier to scroll, and by doing so, user can see that there is indeed a correlation between temperature and vapor content throughout the whole dataset. Hypotheses about possible correlations can be confirmed in detail later on by applying the drill-down mechanism introduced in Section 5.1. In that case, the free space on the right hand side of Fig. 7(b) will again be filled with the structural view of expanded clusters.

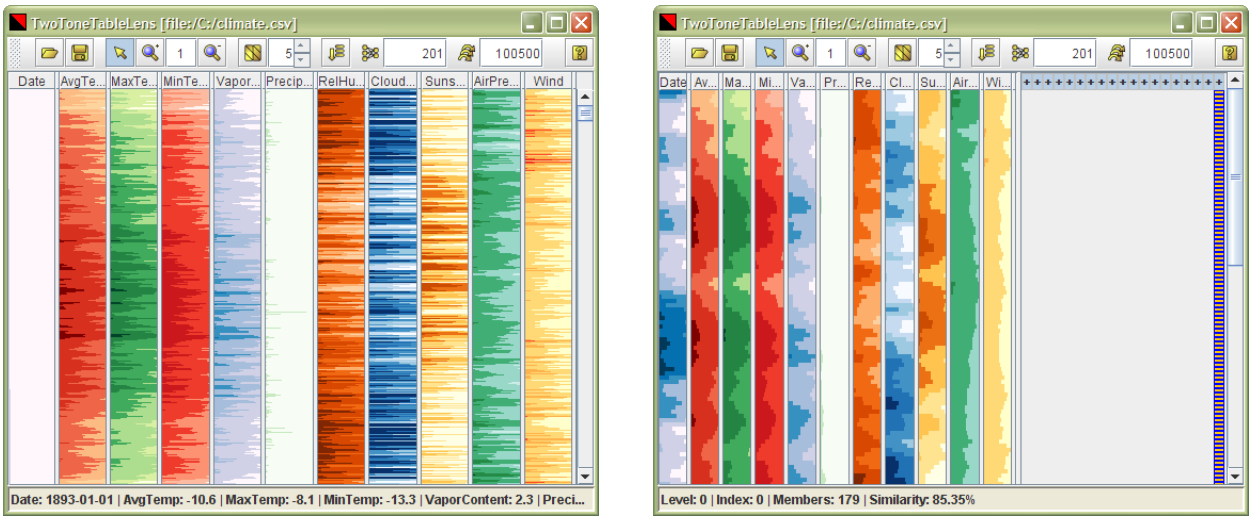

Figure 7. Climate dataset – (a) A correlation between temperature and vapor content (4th and 5th column) can be guessed just from looking at the first 365 (out of 40419) data vectors; (b) Applying clustering and collapsing the entire cluster hierarchy helps in confirming the correlation.

5.3 Expert Experience Report

In this section, we report on an initial attempt to evaluate our approach by informal expert interviews. In particular, we gave our tool to the expert biologist of our graduate school, Ben. Ben is interested in our approach since he needs visual support for his data analysis tasks. In one of his experiments he tested the genetic reaction of immune cells on two different parameters of environmental influence. The resulting data spanned a table of 14 columns (regulation values) and over 15.000 rows (genes). After a brief introduction to our tool Ben needed a little time to get used to the Two-Tone representation. Therefore, he modified the number of color segments, starting with the TableLens representation (one interval) and then stepwise increasing it to five. Once feeling confident, Ben applied hybrid clustering to the data and entirely collapsed the Icicle Plot for a general overview. Ben was mostly interested in genes that are specifically upregulated for one parameter. As it turned out, the SOM clustering generated exactly two clusters with genes that fulfill this requirement. To gain more confidence Ben expanded the Icicle Plot and concentrated on the respective values. He pointed out two very high values as being incorrect. He adjusted the color scale segment borders such that there was one large segment for the incorrect outliers and several smaller ones for the other values, which thus could be read off more precisely.

In Ben's opinion the most helpful feature of the extended TableLens is that many rows but also many columns

can be explored at the same time while still being able to read off specific values. He said, that the clustering facilitates the search for interesting data elements. Additionally, he suggested that functionality for extracting table parts into new tables should be introduced. Thus, he could explore and sort different table parts separately.

6. SUMMARY & FUTURE WORK

Our work shows that the Table Lens can be subject to visual and analytical extensions improving the usefulness of the original approach. We proposed two major extensions. The first is the integration of Two-Tone Pseudo Coloring (TTPC) as an alternative visual encoding. In that context, we presented a heuristic for color scale segmentation that enables users to read precise values more easily. This heuristic can also be applied to other color-based visualizations. The second extension is a hybrid clustering algorithm that helps to reduce strong oscillation in the representation and to communicate the underlying structure of the data. Strong oscillation is reduced by automatic rearrangement of data vectors with respect to clustering results. The exploration of underlying structure is facilitated with the help of an Icicle Plot that represents generated clusters. Additionally, a drill-down mechanism (collapsing and expanding Icicle Plot elements) provides overview and detail on demand.

In future work, we plan to address the following issues related to the visualization time-oriented data. Our current approach to resolve strong data oscillation sacrifices the natural order in time-series. A first idea to solve that problem is to omit rearrangement of data vectors, but instead extend the heuristic for color scale segmentation. The goal is to succeed in finding a segmentation such that both the segmentation is expressive and the number of crucial color switches is reduced. Secondly, since people are used to look at horizontal time axes, we plan to implement a flipping functionality such that the table can be rotated by 90 degrees counterclockwise. This would provide the option to switch between a table and a chart-like representation on demand.

It is planned to achieve these goals in cooperation with Ben and other expert biologists. A large set of biological wet lab data will be available, which provides ample opportunity to apply, evaluate, and refine our Table Lens extensions.

ACKNOWLEDGMENTS

We would like to thank Prof. Takafumi Saito from Tokyo University of Agriculture and Technology for providing the original version of Figure 1.

REFERENCES

1. S. K. Card, J. D. Mackinlay, and B. Shneiderman, eds., *Readings in Information Visualization: Using Vision to Think*, Morgan Kaufmann, San Francisco, 1999.
2. R. Rao and S. K. Card, "The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus + Context Visualization for Tabular Information," in *Proceedings of ACM Conference on Human Factors in Computing Systems (CHI'94)*, ACM Press, 1994.
3. T. Saito, H. N. Miyamura, M. Yamamoto, H. Saito, Y. Hoshiya, and T. Kaseda, "Two-Tone Pseudo Coloring: Compact Visualization for One-Dimensional Data," in *Proceedings of IEEE Symposium on Information Visualization (InfoVis'05)*, IEEE Press, 2005.
4. C. A. Brewer, "Color Use Guidelines for Mapping and Visualization," in *Modern Cartography*, A. M. MacEachren and D. R. F. Taylor, eds., Elsevier Science, Tarrytown, 1994.
5. L. D. Bergman, B. E. Rogowitz, and L. A. Treinish, "A Rule-based Tool for Assisting Colormap Selection," in *Proceedings of IEEE Visualization (Vis'95)*, IEEE Press, 1995.
6. C. Tominski, P. Schulze-Wollgast, and H. Schumann, "Enhancing Visual Exploration by Appropriate Color Coding," in *Proceedings of International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision (WSCG'05)*, UNION Agency, 2005.
7. C. Ware, *Information Visualization: Perception for Design*, Morgan Kaufmann, San Francisco, 2000.
8. A. Telea, "Combining Extended Table Lens and Treemap Techniques for Visualizing Tabular Data," in *Proceedings of Joint Eurographics - IEEE VGTC Symposium on Visualization (EuroVis'06)*, Eurographics Association, 2006.

9. M. A. Harrower and C. A. Brewer, "ColorBrewer.org: An Online Tool for Selecting Color Schemes for Maps," *The Cartographic Journal* **40**(1), 2003.
10. M. Kreuseler and H. Schumann, "A Flexible Approach for Visual Data Mining," *IEEE Transactions on Visualization and Computer Graphics* **8**(1), 2002.
11. T. Kohonen, *Self-organizing Maps*, Springer, Berlin, 2001.
12. L. Kaufmann and P. J. Rousseeuw, *Finding Groups in Data – An Introduction to Cluster Analysis*, Wiley-Interscience, New York, 1990.
13. T. Barlow and P. Neville, "A Comparison of 2-D Visualizations of Hierarchies," in *Proceedings of IEEE Symposium on Information Visualization (InfoVis'01)*, IEEE Press, 2001.
14. M. John, C. Tominski, and H. Schumann, "Prototype: Two-Tone Pseudo Colored TableLens." `http://www.informatik.uni-rostock.de/~ct/TTTL.html` (accessed November 2007), 2006.
15. E. Anderson, "The Irises of the Gaspé Peninsula," *Bulletin of the American Iris Society* **59**, 1935.
16. T. Nocke, H. Schumann, and U. Böhm, "Methods for the Visualization of Clustered Climate Data," *Computational Statistics* **19**(1), pp. 75–94, 2004.