Eurographics/ IEEE-VGTC Symposium on Visualization (2007)
Ken Museth, Torsten Möller, and Anders Ynnerman (Editors)

See what you know: analyzing data distribution to improve density map visualization

Enrico Bertini [1,2], Alessio Di Girolamo[1], Giuseppe Santucci[1]

[1] Dipartimento di Informatica e Sistemistica - Università di Roma "La Sapienza"
[2] Department of Informatics, University of Fribourg - Switzerland

Abstract

Density maps allow for visually rendering density differences, usually mapping density values to a grey or color scale. The paper analyzes the drawbacks arising from the commonly used strategies and introduces a novel technique able to improve the overall mapping process. The technique is driven by statistical knowledge about the density distribution and a set of quality metrics allows for validating, in an objective way, its effectiveness.

Categories and Subject Descriptors (according to ACM CCS): I.3.6 [Computer Graphics]: Methodology and Techniques I.4.3 [Enhancement]: Grayscale manipulation

1. Introduction

Density maps allow for visually rendering density differences. A widely used solution is to represent density values through a grey or color scale, either using a linear mapping [JS98] or a more effective non linear mapping [HMM00].

The paper analyzes the drawbacks arising from the commonly used strategies and introduces a novel technique able to improve the overall mapping process, presenting the user with an increased number of visual density differences. The technique is driven by statistical knowledge about the density distribution and a set of quality metrics allows for validating, in an objective way, its effectiveness.

The framework defined in the paper is used to generate grey or color scale density maps for 2D scatter plots; however it is the authors' believe that the method is general enough to be used in the assignment of any finite visual attribute, like size or line thickness, to a set of data values.

The paper is structured as follows: Section 2 deals with related proposals, Section 3 formalizes the problem, Sections 4 and 5 point out the problems arising from conventional linear and non linear mappings, respectively. Section 6 describes our proposal. Section 7 provides quality metrics that are used in the case study described in Section 8 to validate the overall framework. Finally, Section 9 concludes the paper, outlining open issues and future work.

2. Background and Related work

Density visualization is a well known topic in infovis and the traditional way of representing density through shades of grey or color is shared by several proposals.

Information Mural [JS98] creates a mapping between pixels of a source space, with higher resolution, to pixels of a reduced target space with lower resolution. Each target pixel represents multiple pixels of the original space and data overlapping are linearly mapped to color intensity.

Van Liere et al. in [vLdL03] propose "graph splatting", a technique to represent density with 2D scalar fields. A continuous variation in density is obtained splitting the screen in small cells and positioning in the center of each a gaussian 2D function. The resulting visualization is obtained by summing up the contribution of all functions. The result is a continuous grey cloud of different shades of grey that permits to distinguish between areas of high and low density. The same technique is also used in a number of other proposals like in [CJM04], to detect regions of interest in a scatter plot, and in [Bec97, Yan00] where it is applied to 3D scatter plots.

Artero et al. in [AdOL04] provide an effective solution for parallel coordinates. They use kernel-based density estimation [Sil90] to map density of pairs of data dimensions to the intensity of line segments between parallel coordinates. The method permits to detect and isolate clusters on parallel coordinates. A similar approach is used in [JLJC05] where the user can, in addition, control a transfer function to provide non-linear mapping between density and pixel intensity.

These techniques represent a broad spectrum of methods that can be applied on a variety of visualizations. All of them, however, do not scale well in case of highly skewed/long-tailed density functions: many elements share low density values and few elements share very high density values. In this case the visualization loses its dynamic range and all visual items but few ones get colored with low values.

There are two related techniques specifically dealing with this problem: Herman et al.'s mapping through density functions [HMM00] and *histogram equalization*. Herman et al. argue that an effective mapping can be obtained only by taking into account data distribution. They analyze the data frequency distribution and use the corresponding density function (obtained by integration) to map the data values to the target values, producing a higher dynamic range.

Even if not explicitly acknowledged by the authors, the technique resembles very closely *histogram equalization* [PAA*87] [Rus95], a well known image processing technique used to enhance images with poor dynamic range. The histogram represents the number of pixels shared by each grey value and the equalization produces a new histogram in which the bars are spread along the whole spectrum. The theoretical background is exactly the same as in density mapping, the equalization is obtained using the probability density function as a transfer function. The only difference between the two is that in the former the method is applied on the data domain, in the latter in the device domain.

Our work takes direct inspiration from these techniques. We start from them to create pixel-based [Kei00] effective density maps, in which the dynamic range of data points is increased. In our examples we use the *HSI color model* [KK95] that permits to span across the whole color spectrum and, at the same time, to follow a continuously increasing/decreasing intensity value. The continuous and monotonic behavior of the scale is a key factor to represent ordered values through color shades [War00].

We also acknowledge the need of taking into account data density and provide amelioration over existing techniques allowing for automatically creating effective data overviews of skewed data set distributions.

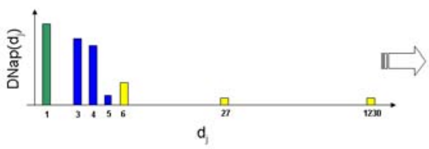
3. Definitions

The problem we are dealing with corresponds to mapping a data feature (showing N_{DV} distinct values) to a finite (N_L distinct levels) visual attribute that preserves the ordering of values. The mapping is computed using the data value frequency distribution and pursuing the goal of visually preserving as many data differences as possible. In the following the problem is formalized for the generation of 2D density maps utilizing the *HSI color model* [KK95] and using a color scale obtained by linear interpolation of two (high brightness, low brightness) points in the HSI space. It means that the used scale has a monotonically increasing brightness and that it uses a full color range. In the rest of the paper we will refer to color values as points of an ordered axis corresponding to the HSI interpolating line. We split the axis in 255 values as a way to make the results comparable in terms of the traditional grey scale. We use a color scale in the examples because the images are more effective but all following considerations apply to a grey scale as well.

Assume we are plotting n data points on a 2D scatter plot. Because of data items with same values and rounding issues, some pixels host more than one data point, i.e., each pixel is characterized by a specific *density*. For a given data set plotted on a given area the scatter plot exhibits N_{DV} distinct not null density values, $d_1, \ldots, d_{N_{DV}}$ that are mapped to a $N_L = 255$ levels color scale. A mapping is characterized by the $ColorCode(d)$ function that assigns a color tone to each d_i; in general, not all available N_L tones are used and we denote with N_{UL} the number of used levels ($N_{UL} \leq N_L$).

N_{AP} denotes the number of pixels whose density is greater than 0. $DN_{AP}(d)$ computes how many pixels share the density d. Similarly, $CN_{AP}(c)$ computes how many pixels share the color value c. Note that, while $\sum_{i=1}^{N_{DV}} DN_{AP}(d_i) = \sum_{j=1}^{N_{UL}} CN_{AP}(c_j) = N_{AP}$, because of *density collisions* [†], for each different density d_i it holds that $DN_{AP}(d_i) \leq CN_{AP}(ColorCode(d_i))$.

Figure 1 illustrates an example in which $N_{DV} = 7$ and $N_{UL} = 3$. Density values range from $d_1 = 1$ to $d_7 = 1230$ and bar colors correspond to the three color tones ($c_1 = 63$, $c_2 = 127$, and $c_3 = 255$) that have been assigned to each d_i through the $ColorCode$ function; because $N_{DV} \geq N_{UL}$ some color values represent more than one density (e.g., $ColorCode(d_2) = ColorCode(d_3) = ColorCode(d_4) = 127$: three densities collide on the same color tone).



d_j	c_i
1	63
3	127
4	127
5	127
6	255
27	255
1230	255

Figure 1: *Mapping example.*

Summarizing, in the rest of the paper we use the following notation:

- N_{DV} (Number of Distinct Values), the number of distinct data values;
- N_L (Number of Levels), the number of distinct visual attribute values;

[†] In this paper the term collision (or density collision) denotes that two different density values are represented by the same color level

- N_{UL} (Number of Used Levels), the number of visual attribute values used in the current mapping;
- $ColorCode(d)$, a function assigning a color tone $c \in N_L$ to a density $d \in N_{DV}$;
- N_{AP} (Number of Active Pixels), the number of pixels hosting at least one data item;
- $DN_{AP}(d)(DensityN_{AP})$, a function returning the number of pixels sharing density d;
- $CN_{AP}(c)(ColorN_{AP})$, a function returning the number of pixels sharing color tone c.

4. Linear mapping

A linear mapping is defined by the following function:

$$ColorCode(d) = Round(N_L(d - d_{min})/(d_{N_{DV}} - d_{min}))$$

where d_{min} is the lowest density value. This intuitive approach works well only uniform distributions. As an example, Figure 2 shows a linear mapping to a $N_L = 255$ color scale of the 2005 Infovis contest data set, 2002 USA companies, on a 800×450 2D scatter plot built using their longitude and latitude. The skewed company density frequency distribution is depicted in Figure 3, showing for each d_i the number of pixels sharing that value, $DN_{AP}(d_i)$ (X axis uses a logarithmic scale).

Figure 2: *Infovis'05 contest linear mapping.*

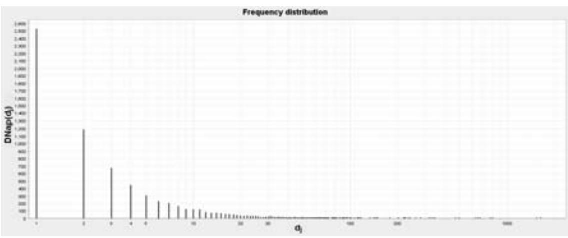


Figure 3: *Infovis'05 contest density frequency distribution.*

$N_{DV} = 126$ (i.e., the scatter plot contains 126 different not null densities), $d_1 = 1$ while $d_{N_{DV}} = 1633$ (i.e., the maximum density is 1633 companies per pixel). Looking at the frequency distribution it is quite evident that most active pixels host few companies. As an example, $DN_{AP}(1) = 2526$ (i.e., 35.87% of active pixels contain just one company), $DN_{AP}(2) = 1182$ (i.e., 16.79% of active pixels contain two companies); on the other hand, $DN_{AP}(1633) = 1$, i.e., just one pixel contains 1633 companies.

As a consequence, the linear mapping of Figure 2 hides some useful information because:

1. most density differences are hidden: e.g., low density values (1..10) are represented with similar and low color values, 1 and 2, and, as a consequence, 84% of active pixels are not distinguishable;
2. the frequency distribution being very spare, most color tones correspond to empty zones, so few of the available N_L values are used, i.e., $N_{UL} = 46$.

Summarizing, the linear mapping fails when dealing with non uniform frequency distributions for three reasons:

1. it assigns to most of the pixels very low color scale values and to few elements high color scale values;
2. it assigns to most of the pixels very similar (if not the same) color scale values, making them hardly distinguishable;
3. a large number of color scale values are not used in the visualization (82%, in the above example).

It is worth noting that the non uniform behavior of the frequency distribution presented above, is not an exception of the specific example but a common case in data analysis: we tested different real data sets and the large majority exhibited similar frequency distributions.

5. Non linear mapping

In order to overcome the drawbacks of linear mapping, several alternatives have been proposed; the most interesting ones, described in the following, share the idea of using the density function as a means to compute the mapping between density values and color tones. In the following we discuss the underlying ideas and the yet unsolved problems.

5.1. Density function driven mapping

Herman et al. [HMM00] propose the use of the density function in the context of graph drawing, but their analysis is general enough to be applied to our case of density maps. Still considering the Infovis data set example, we can see in Figure 4 the density function, obtained integrating the frequencies shown in Figure 3.

The density function is used as a transfer function, mapping each density value d_i to a color value proportional to the percentage of pixels that share a density equal or less than d_i. More formally:

$$ColorCode(d_j) = Round(N_L \sum_{i=1}^{j} DN_{AP}(d_i)/N_{AP})$$

As an example, Figure 2 contains 7042 active pixels ($N_{AP} = 7042$) and the pixels containing just one company

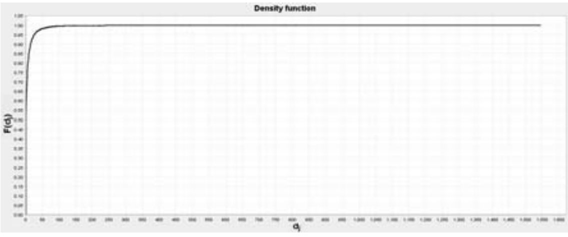


Figure 4: *Infovis'05 contest density function.*

Figure 5: *Infovis'05 contest density function mapping.*

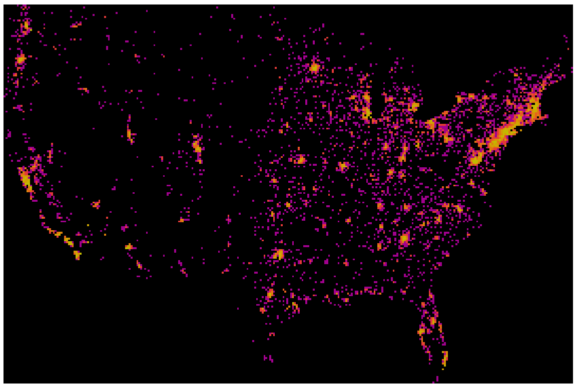


Figure 6: *Infovis'05 contest histogram equalization mapping.*

(2526) are mapped to color value 91 (255*2526/7042); pixels containing two elements (1182) are mapped to color value 134 (255*(2526+1182)/7042) and so on. The result of such a mapping is visible in Figure 5 and, even if it is better than the one produced by linear mapping, it still exhibits three main drawbacks:

1. the lowest adopted color tone, c_1, is set only considering the $DN_{AP}(d_1)$ percentage, disregarding the highest density value $d_{N_{DV}}$, and the total amount of density values being represented, N_{DV}. That usually produces a too high c_1 value, leaving few available color tones and thus using a limited color scale range. In the example, $c_1 = 91$ and $c_2 = 134$: the mapping, after the two first assignments, has used more than half of the scale, while there are still $N_{DV} - 2 = 126$ remaining densities to assign;
2. each density d_i that is shared by less than N_{AP}/N_L pixels (i.e., $DN_{AP}(d_i) \leq N_{AP}/N_L$) will likely be assigned to the same color value as d_{i-1}, thus producing a collision and hiding potentially useful visual clues from the end user, even if there are unused color values. As an example, the 10 density values in the interval [53,62] are mapped on the *same* color value, 251, while the overall mapping uses only 39 color tones. The same holds for the highest 14 densities, ranging from 280 to 1633, which are mapped on the same color tone, 255;
3. the highest density values are mapped to similar and unnecessarily high color tones. As an example, the highest 79 density values in the interval [48, 1633] are mapped to the 6 color values in the interval [250,255] producing a high number of density collisions.

5.2. Histogram equalization

Histogram equalization is a well known technique used in image processing and is very similar to the density function mapping described above. Starting from the *color scale* frequency distribution produced by the linear mapping, it alters the color assignment re-mapping each color value c_i to a new color value, which is proportional to the percentage of pixels sharing a color value equal or less than c_i. More formally:

$$HE(c_j) = Round(N_L \sum_{i=1}^{j} CN_{AP}(c_i)/N_{AP})$$

The visual result of the HE mapping is shown in Figure 6; the image cumulates the drawbacks produced by both linear and density function mappings, increasing the number of density collisions and reducing the number and the range of used color tones: in the example N_{UL} decreases from 43 to 13, spanning on the color range [174,255].

6. Improving density mapping

To overcome the drawbacks of both linear and non linear mappings it is necessary to take into account that:

- the density frequency function is not uniform;
- N_{DV} is a finite and small number;
- the assignment function targets a visual attribute characterized by a finite and small range of values ($N_L = 255$ color levels).

The proposed mapping technique takes into account all these parameters, carefully mapping density values to color values using a two steps procedure (conceptually similar to the two abstract steps proposed by Herman et al. in [HMM00]):

1. splitting the frequency distribution X axis in $k \leq max(N_{DV}, N_L)$ intervals;
2. assigning to each interval a color scale value.

From this point of view the linear mapping corresponds to splitting the X axis in N_L equal intervals, assigning to each interval i a color value i. The rationale behind it is to use color scale values proportional to density values. We instead follow the basic idea of non linear mappings. In order to increase the number of perceivable density differences, it is better to bear several kind of distortions and lose the intensity of density differences than having a flat representation. The final image is more useful as an overview because it is easier to spot interesting regions. That corresponds, as detailed in the next sections, to splitting the X axis in *non equal* intervals and to use color scale values as distant as possible. This second issue is managed during the second step: if $k < N_L$ the mapping chooses k color scale values, maximizing their distance and disregarding the density values they represent; if $k = N_L$ the mapping uses all N_L values.

6.1. Splitting strategies

In the following it is assumed that the second step is performed according to the general principle described so far, i.e., maximizing the distance among the used color values.

Two main cases hold: (1) $N_{DV} \leq N_L$ or (2) $N_{DV} > N_L$.

$N_{DV} \leq N_L$ **- One to one mapping**. In this case the algorithm splits the X axis in N_{DV} intervals, each of them containing one density value. In the Infovis contest example it corresponds to using 126 distinct color scale values vs the 43 used in the linear mapping, the 39 used in the density function mapping, and the 13 in the histogram equalization mapping. Most importantly, all lower density values are represented by different color scale values and the visual differences among low density values increases. The result is presented in figure Figure 7. Low density values are now better visible, together with the higher ones.

Figure 7: *Infovis'05 contest one to one mapping.*

$N_{DV} > N_L$ **- Uniform color scale frequency distribution algorithm**. All data sets used in our experiments presented a $N_{DV} < N_L$; however we discuss this case for (a) having a general strategy and (b) dealing with color scale characterized by a deliberately reduced number of values, in order to present the user with clearly distinguishable tones. In this case, some intervals contain more than one density value. In order to maximize the visible density differences we use an algorithm that tries to assign to each color tone the *same* number of pixels, producing a uniform color scale frequency distribution. To obtain such a result, starting from the density frequency distribution (Figure 3), we split the X axis in a series of intervals, each one containing a number of pixels as close as possible to N_{AP}/N_L. Because we are working with discrete values, we cannot guarantee that the average value is always N_{AP}/N_L and, as a consequence, we use an algorithm that reduces the variance through the analysis of *peaks*, those elements for which $DN_{AP}(d_i) \geq N_{AP}/N_L$.

The algorithm sketch is the following:

1. Set $N_v = N_L$, $N_p = N_{AP}$ and the interval size (threshold) $T = N_p/N_v$;
2. Compute the density frequency distribution DFD and sort it according to $DN_{AP}(d_i)$ values;
3. Scan DFD in a descending fashion and whenever $DN_{AP}(d_i) > T$
   - mark d_i as a *peak*;
   - set $N_v = N_v - 1$ and $N_p = N_p - DN_{AP}(d_i)$;
   - set $T = N_p/N_v$
4. Analyze DFD in order of density values and create a new interval each time that:
   - a d_i marked as a peak is found, or
   - the sum of the encountered $DN_{AP}(d_j)$ values is equal or greater than T.

The algorithm presents three different complexities:

1. constructing the density frequency distribution: O(n), where n is the number of plotted points;
2. sorting the different $DN_{AP}(d_i)$ values: $O(N_{DV}\ log_2(N_{DV}))$;
3. deriving the k intervals: O(N_{DV}).

Complexity is governed by the first step (typically, $n >> N_{DV}$), thus it is linear in the number of data elements.

Summarizing, our technique improves the mapping process using these three main strategies:

1. it uses as many color tones as possible;
2. it maximizes the distance among used tones;
3. when density collisions are unavoidable it tries to uniformly distribute collisions among the used color values, producing a color frequency distribution as uniform as possible.

The effectiveness of this mapping strategy is demonstrated through a case study in Section 8 using the quality metrics described in the next section.

Figure 8: *Infovis'05 contest uniform color scale mapping.*

7. Quality Metrics

To evaluate the mapping strategy presented in the paper we define three quality metrics.

Metric 1 (ColorScale Usage)

Description: One of the objectives of a good mapping strategy is to use a number of color levels as great as possible: this metric indicates how effectively the mapping uses the color scale levels, comparing N_{UL}, indicating how many levels the mapping is using with the maximum number of levels the algorithm could use with the actual data set, $\min(N_{DV}, N_L)$.

Equation:

$$CSU = \frac{N_{UL}}{\min(N_{DV}, N_L)} \tag{1}$$

Range: $[0,1]$, **Objective:** $CSU = 1$

Metric 2 (Color Scale Active Range)

Description: Another mapping objective is to use the full range of the color scale values: this metric indicates how much of available color range the mapping is using, comparing the used range with the full color scale range.

Equation:

$$CsAR = \frac{c_{N_{UL}} - c_1}{c_{max} - c_{min}} \tag{2}$$

Range: $[0,1]$, **Objective:** $CsAR = 1$

Metric 3 (Color Separation)

Description: In order to present an increased number of visual density differences another objective is to uniformly distribute color tones across the color scale range. This metric deals with this issue comparing the minimum distance between two adjacent tones with the maximum one.

Equation:

$$CS = \frac{\min(c_j - c_{j-1})}{\max(c_j - c_{j-1})}, \forall j \in \{2, \ldots, N_{UL}\} \tag{3}$$

Range: $[0,1]$, Objective: $CS = 1$

8. Case Study

In this section the paper proposal is validated using the quality metrics described in the previous section. We use a different data set containing about 150,000 mail parcels, plotted on 250×250 pixels according to their weight (X axis) and volume (Y axis). That produces $N_{DV} = 174$ different densities with a maximum value of 781. Figure 9 shows the density frequency distribution that is quite similar to the one shown in Figure 3: many pixels share few low density values, and few pixels span the rest of the density range.

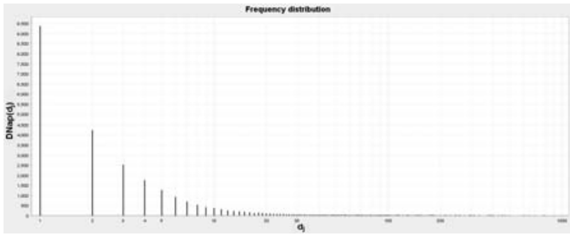


Figure 9: *Parcels density frequency distribution.*

In *linear mapping* (Fig. 10 (a)) almost all pixels are clamped into a narrow range of low color levels. The visual effect is clear: all but few pixels share very similar and low color values, and thus many density differences are lost. The quality metrics confirm the visual effect: while color tones cover all color scale range (CsAR=1) they are few (CSU=0.529) and non uniformly distributed (CS=0.022). In *density function mapping* the image improves because the algorithm takes into account the distribution of pixels and assigns a higher color value where pixels are more numerous, that is, in the lower section of the scale. As we noted above, however, the algorithm wastes a high number of color tones (CSU=0.184) that are completely unused and starts from a too high level reducing the used color scale range (CsAR=0.62). The consequences are visible in the image: the distance among low color tones increased but many elements are black shaded and are non distinguishable (CS=0.023). In *one to one mapping*, instead, each density has its own color level (CSU=1), color levels are more uniformly distributed (CS=0.5), and the whole scale is used (CsAR=1). The image presents a higher number of density differences therefore its inspection allows for extracting more information than the previous cases.

The fourth image deals again with the problem of how many color levels a human is able to distinguish. Unfortunately, for our specific case a single value cannot be drawn because the perception of light intensity is extremely dependent from contextual factors: ambient light, lightness of surrounding elements, object's size, etc. [War00]. Levkowitz et al. in [Lev96] suggest a range between 60 and 90 JNDs in greyscales while color scales present a much higher number.

It is our belief, however, that the size of the objects plays a major role and in our informal experiments, where the object are small pixels, we concluded that even a range between 60 and 90 for color scale is too large and that a value around 20 and 30 levels is more realistic.

A nice feature of our proposed algorithm, however, is that its execution does not depend on the number of available levels. As long as the number of available levels is higher than the number of densities it applies a one to one mapping. As soon as this condition does not hold, it performs a *uniform color scale* mapping, optimizing the density collisions. In both cases, the whole color scale is always used without waste, the distance between assigned color values is maximized and, as a consequence, the number of distinct densities is also maximized.

Figure 10 (d) shows the result of this last kind of mapping where the number of available color levels has been reduced to 30. The histogram shows a reduced set of levels and the effects can be seen in the image. Comparing this last visualization with the previous ones allows for better understanding of the algorithm effects. All 30 color levels are used in the mapping (CSU=1), covering the full scale range (CsAR=1); densest areas are still very dark but they are surrounded by brighter areas thus showing more differences (CS=0.889). At the lower end of the scale, low density pixels are still visible and areas with different density can be perceived.

9. Conclusion And Future Work

The paper presented a novel approach to map data density to a color scale, extending the classic non linear mappings taking into account all details of the density frequency distribution and pursuing the goal of presenting the user with as many density differences as possible. The proposal is quite general and can be extended to other cases where a mapping between a skewed data dimension and a visual attribute is required.

We are planning to extend and refine our work toward several directions:

- challenge the generality of the approach applying it to a broader class of visual attributes, e.g., size and thickness;
- better investigate and formalize the similarity of 2D scatter plot density frequency distribution;
- provide some semi-automatic environment to assist the user during the interactive data exploration, allowing end users to apply different mapping strategies on demand.

10. Acknowledgements

Work supported by DELOS Network of Excellence.

References

[AdOL04] ARTERO A. O., DE OLIVEIRA M. C. F., LEVKOWITZ H.: Uncovering clusters in crowded parallel coordinates visualizations. In *Proc. of IEEE Symposium on Information Visualization* (2004), IEEE Computer Society, pp. 81–88.

[Bec97] BECKER B. G.: Volume rendering for relational data. In *Proc. of the IEEE Symposium on Information Visualization* (1997), IEEE Computer Society, p. 87.

[CJM04] CHIRICOTA Y., JOURDAN F., MELANCON G.: Metric-based network exploration and multiscale scatterplot. In *Proc. of the IEEE Symposium on Information Visualization* (2004), IEEE Computer Society, pp. 135–142.

[HMM00] HERMAN I., MARSHALL M. S., MELANÇON G.: Density functions for visual attributes and effective partitioning in graph visualization. In *Proc. of the IEEE Symposium on Information Vizualization* (2000), IEEE Computer Society, p. 49.

[JLJC05] JOHANSSON J., LJUNG P., JERN M., COOPER M.: Revealing structure within clustered parallel coordinates displays. In *Proc. of IEEE Symposium on Information Visualization* (2005), IEEE Computer Society, pp. 81–88.

[JS98] JERDING D. F., STASKO J. T.: The information mural: A technique for displaying and navigating large information spaces. *IEEE Transactions on Visualization and Computer Graphics 4*, 3 (1998), 257–271.

[Kei00] KEIM D. A.: Designing pixel-oriented visualization techniques: Theory and applications. *IEEE Transactions on Visualization and Computer Graphics 6*, 1 (2000), 59–78.

[KK95] KEIM D. A., KRIEGEL H.-P.: Issues in visualizing large databases. *Proc. of the IFIP working conference on Visual database Systems* (1995), 203–214.

[Lev96] LEVKOWITZ H.: Perceptual steps along pseudocolor scales. *International Journal of Imaging Systems and Technology 7*, 1 (1996), 97–101.

[PAA*87] PIZER S. M., AMBURN E. P., AUSTIN J. D., CROMARTIE R., GESELOWITZ A., GREER T., ROMENY B. T. H., ZIMMERMAN J. B.: Adaptive histogram equalization and its variations. *Comput. Vision Graph. Image Process. 39*, 3 (1987), 355–368.

[Rus95] RUSS J. C.: *The image processing handbook (2nd ed.)*. CRC Press, Inc., Boca Raton, FL, USA, 1995.

[Sil90] SILVERMAN B.: *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, 1990.

[vLdL03] VAN LIERE R., DE LEEUW W.: Graphsplatting: Visualizing graphs as continuous fields. *IEEE Transactions on Visualization and Computer Graphics 09*, 2 (2003), 206–212.

[War00] WARE C.: *Information visualization: perception for design*. Morgan Kaufmann Publishers Inc., 2000.

[Yan00] YANG L.: Interactive exploration of very large relational datasets through 3d dynamic projections. In *Proc. of the ACM SIGKDD* (2000), ACM Press, pp. 236–243.

(a)

Linear

CSU =0.529
CsAR= 1
CS = 0.022

(b)

Density Function

CSU =0.184
CsAR= 0.62
CS= 0.023

(c)

One to one

CSU =1
CsAR= 1
CS = 0.5

(d)

Un. color scale

CSU =1
CsAR= 1
CS = 0.889

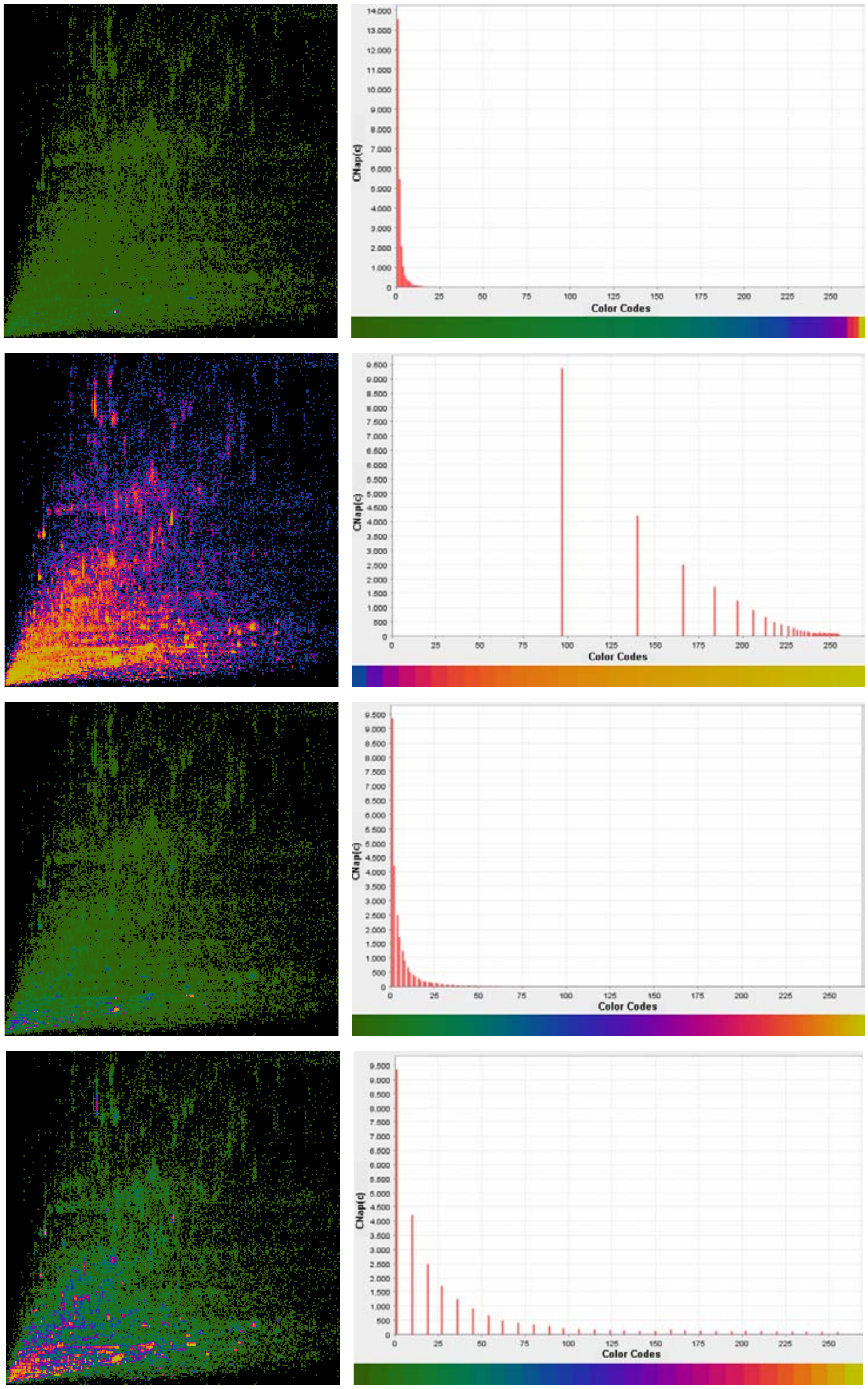


Figure 10: *Mapping algorithms applied to the parcels data set: (a)linear mapping; (b)density function mapping; (c)one to one mapping; (d)uniform color scale mapping with a set of color tones reduced to 30 perceptible steps.*