

Is it darker? Improving density representation in 2D scatter plots through a user study

Enrico Bertini and Giuseppe Santucci

Dipartimento di Informatica e Sistemistica - Università di Roma “La Sapienza”
Via Salaria, 113 - 00198 Rome, Italy

ABSTRACT

Density differences are one of the main features users perceive in 2D scatter plots. However, because of pixels' collisions, some areas become saturated and such differences are lost. To solve this problem, several proposals rely on sampling the dataset before visualizing it. Some of these introduce precise measures to understand the image degradation and use numerical differences in pixels to estimate density differences. It is our opinion that this issue deserves a deeper analysis, taking into account perceptual issues. In this paper we describe a study we conducted to understand the relationship between numerical pixel density and the perceived density. The results obtained were used to refine a sampling technique we developed to preserve relative densities in the context of 2D scatter plots

Keywords: visual clutter, quality metrics, data sampling, user studies.

1. INTRODUCTION

Visualizing data on 2D representations that permit graphical elements to overlap, often results in cluttered images where many pixels become over plotted and, consequently, relevant features are hidden or distorted. In this paper we focus on bi-dimensional scatter plots that are one of the most widely adopted visual technique and that are heavily affected by this problem.

Some proposals coping with this issue exist. The main techniques consist in either rearranging the pixels on the screen¹ or introducing some form of sampling, to reduce the number of pixels on the screen.^{2,3} Recently, we proposed a formal frameworks to estimate the amount of degradation resulting from a given visualization, measuring the visual clues that are hidden from the user and sampling the data to enhance the representation.

We proposed a quality metric able to count the density differences that are lost because of pixel over-plotting. Such a quality metric is based on the idea of comparing the number of active pixels (i.e., turned on pixels) in order to estimate density differences and to guide an automatic sampling technique. In this paper this result is extended, introducing the point of view (literally) of the user through a user study, assessing the real pixel differences necessary to produce a visual clue of density difference. The results we obtained from the user study have been included in our formal framework, tuning our algorithms and producing more readable images.

The contribution of the paper is twofold:

1. it presents a user study that allows for understanding, in a numeric way, the user perception of relative density and
2. it incorporates in a formal model such results, providing a framework to rescue relative densities from crowded 2D scatter plots.

The paper is structured as follows: Section 2 analyzes related works, Section 3 introduces the formal framework we use to characterize clutter and density, the quality metric we use to estimate the image degradation, and our sampling technique, Section 4 describes the user study, Section 5 analyzes the study results and discusses its impact on our algorithms, and, finally, Section 6 presents some conclusions, open problems, and future work.

Further author information: E-mail: {bertini, santucci}@dis.uniroma1.it

2. RELATED WORK

Our work focuses on measuring some quantities associated with the actual visualization of some pieces of information, in order to detect flaws in the representation and to perform corrective actions. To this aim, we devised some metrics based on a formal framework and we performed a user study, which permits to tune the involved parameters.

The fact that the field of Information Visualization needs metrics to provide precise indications on how effectively a visualization presents data is well known⁴ and a definitive and strong set of methodologies/tools is still lacking.

First attempts towards this direction come from Tufte that in his "The Visual Display of Quantitative Information"⁵ proposes an interesting set of measures to estimate the "graphical integrity" of bi-dimensional representations of static data. Measures like the *lie factor*, that is the ratio between the size of an effect, as shown graphically, to its size in the data, or yet, the *data density* that takes into account the size of the graphic in relation to the number of data displayed, are examples of his attempt to systematically provide indications about the quality of the displayed image. Tufte's work, however, applies to paper based 2D visualizations and is expressed more in form of guidelines than in form of precise formulas to use in practice. Brath, starting from the Tufte's proposal, defines new metrics for static digital 3D images.⁶ He proposes metrics such as *data density* (number of data points/number of pixels) that recall Tufte's approach. He provides metrics aiming at measuring the visual image complexity like the *occlusion percentage*, that is the number of the elements occluded in the visual space (having interesting connections with our work), or the *number of identifiable points*, that is the number of visible data points whose position is identifiable in relation to every other visible data point. These metrics are interesting and are more appropriate for characterizing digital images. However, as stated by the author, they are still immature and, unfortunately, they have not been further developed, so we still lack for a good and strong set of metrics.

Using benchmarks to evaluate the effectiveness of visualization systems is a widely accepted practice. Grinstein proposes some benchmarks to run against predefined datasets in order to evaluate and compare different techniques.⁷ However, these benchmarks are generally aimed at measuring their effectiveness with respect to human performance in carrying on some tasks. Instead, our aim is to measure the accuracy of the visualization while representing the data characteristics w.r.t. some visual insight (e.g., data density). As a consequence, our techniques operate at a lower level, dealing with pixels and data points, to provide measures that can be directly exploited to apply corrective actions. It is worth to note that, on the contrary of the above proposals, we will show how the suggested metrics can be exploited *in practice* to take quantitative decisions about corrective actions to enhance the current visualization.

Since measuring in this context means measuring the perception of visual features, perceptual studies play a key role. Many perceptual studies have been conducted in the past to increase the effectiveness of visual systems and to avoid distortions. Results coming from color theory have been applied in practice in the context of data visualization to select color scales that appropriately reflect the properties of underlying data^{8,9}. Preattentive features (visual features detected by the human eye without cognitive workload) have been exploited in the visualization of multivariate data to allow the users for efficiently detecting visual patterns. Healey, in various proposals, inspected the effectiveness of preattentive features analyzing the interferences among them.¹⁰ Another interesting proposal is that of the "Semantic Depth of Field",¹¹ where the authors propose to exploit the depth of field as a visual feature to present objects at different levels of detail, allowing some objects to pop out in the presented image.

Here we present a perceptual study aiming at inspecting the perception of relative densities in a 2D scatter plot. In particular, as will be further explained in section 4, it is aimed at detecting the threshold values beyond which differences in density are perceived.

3. MODELING VISUAL DENSITY AND CLUTTER

In this section we recall our formal framework³ that aims at modeling the clutter produced by over-plotting data. Here we report a synthesis of the original proposal, while in Section 5 we show how this framework is affected by the user study results.

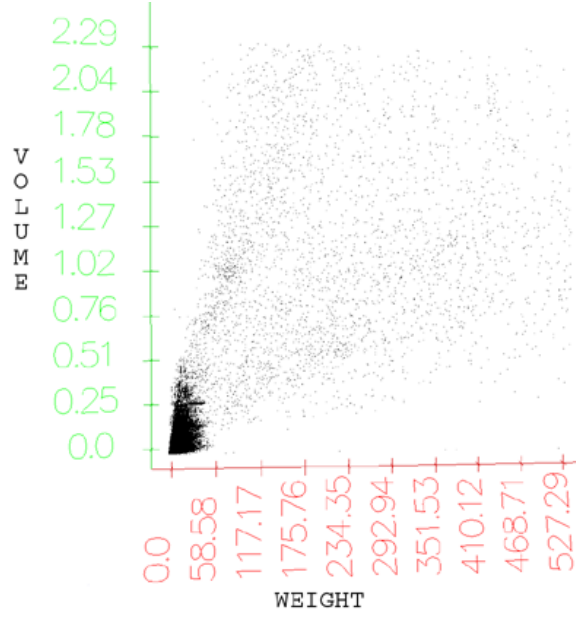


Figure 1. The scatter plot represents a dataset of mail parcels. The parcel's weight is mapped to the X-axis and the parcel's volume to the Y-axis.

We consider a 2D space in which we plot points by associating a pixel to each data element, mapping two data attributes on the spatial coordinates. As an example, Figure 1 shows about 160,000 mail parcels plotted on the X-Y plane according to their weight (X axis) and volume (Y axis). It is worth noting that, even if the number of plotted items is small, the area close to the origin is very crowded (usually parcels are very light and little), so a great number of collisions are present in that area: the most crowded zone contains more than 50,000 parcels (about 30 % of the whole dataset) compressed in less than 1 % of the whole screen.

We derived a function that estimates the amount of colliding points using a probabilistic model based on the following parameters:

- n is the number of points we want to plot;
- p is the number of available pixels;
- k is the number of collisions;
- d is the number of free pixels.

The probability of having *exactly* k collisions plotting n points on an area of p pixels, $Pr(k, n, p)$, is given by the following function:

$$\frac{PERM\left[\frac{p}{n-k} \binom{n-k+k-1}{k}\right]}{p^n} \quad \begin{array}{l} \text{if } n \leq p \text{ and } k \in [0, n-1] \\ \text{or } n > p \text{ and } k \in [n-p, n-1] \end{array}$$

$$0 \quad \text{if } n > p \text{ and } k \in [0, n-p-1]$$

PERM stands for permutations

The basic idea of the formula is to calculate, given p pixels and n plotted points, the ratio between the number of possible cases showing *exactly* k collisions and the total number of configurations.

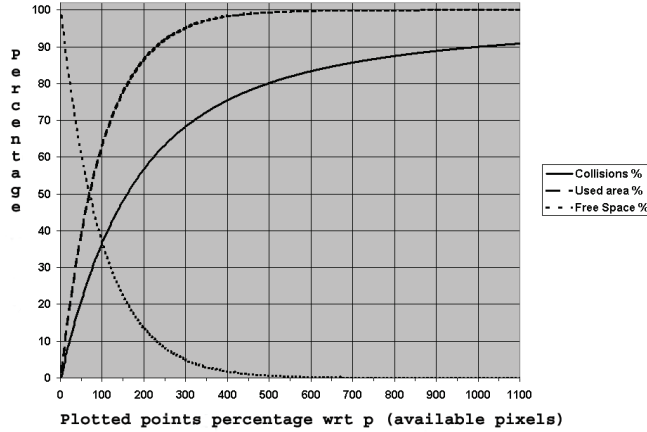


Figure 2. Colliding elements percentage.

The latter is computed considering all the ways in which it is possible to choose n points among p pixels, i.e., dispositions with repetitions: p^n .

To compute the number of configurations with exactly k collisions, we first calculate all the possible ways of selecting $n - k$ non colliding points from p pixels, i.e., combinations *without* repetitions: $\binom{p}{n-k}$. After that, for each of such combinations, we calculate all the possible ways of hitting k times one or more of the $n - k$ non colliding points in order to obtain exactly k collisions, i.e., combinations *with* repetitions: $\binom{n-k+k-1}{k}$. Finally, we count the permutations (PERM) of these combinations.

Note that the above formula is computed under the assumption that the probability of hitting a pixel is the same for all pixels (i.e., $\frac{1}{p}$). While real data do not follow this hypothesis (as the parcels example demonstrates), we can rely on such assumption when considering very small screen areas (see the following subsection).

From the above expression we derived, through a C program, a series of functions (see Figure 2) showing the behavior of the observed area as the number of plotted points grows. The behavior of the functions is quite intuitive: as the number of plotted points n increases the percentage of collisions increases as well while the free space decreases; roughly speaking, we can say that plotting four times the number of the pixels available in the observed zone results in a totally saturated area (1.6% of free space).

We exploit such functions to apply corrective actions, in order to improve the image readability. As an example, assume that we are plotting n points on the area A_1 turning on p_1 pixels and $2n$ points on the area A_2 turning on p_2 pixels. In principle, the user should perceive area A_2 as containing more (i.e., twice as many) points as area A_1 . Because of collisions, $p_2 \leq 2p_1$ and as n increases the user initially loses the information that area A_2 contains twice as many points as A_1 and, for greater values of n , s/he is not able to grasp any difference between A_1 and A_2 . As a numerical example, if we plot 64 and 128 points on two 8×8 areas, the pixels turned on in the two areas will be 41 and 55, so the ratio of displayed pixels is only 1.36. In order to preserve the visual impression that area A_2 contains twice as many points as A_1 accepting a degradation of 20 per cent we have to sample the data (64 and 128 points) as much as 50 per cent resulting in 32 and 64 points that, once plotted, turn on 25 and 41 pixels, i.e., a ratio of 1.6 (20 per cent of degradation).

This example permits to explain the reasons that motivated us to perform the user study. In the above discussion we were trying to present the user with a pixel ratio that resembles, as much as possible, the original data densities and we presented some associated figures. To assess the usefulness of our analysis we should establish how much such numerical differences are perceived as density differences by a user. Therefore, to attack such a problem, we investigated what is the minimum difference in active pixels that still produces a perception of density difference. The idea will be described in detail in Section 4.

3.1. Density and quality metric

The above formal framework gives us a way to control and measure the number of colliding elements; here we give a formal characterization of density and we introduce a quality metric to estimate the degradation of an

image.

We assume the data are displayed on a rectangular screen (measured in inches) and that small squares of area A divide the space in $m \times n$ *sample areas* (SA) where density is measured. Actual figures depends on monitor resolution and size. In the following we assume we are using a monitor of 1280x1024 pixels and size of 13" x10.5" (100 dpi).

Using these figures we have 1,310,720 pixels and if we choose SA of side $l = 0,08$ inch, the area is covered by 20.480 (128x160) sample areas whose dimension in pixel is 8×8 . We consider small areas because it makes the probabilistic assumption used in Section 3 quite realistic.

For each $SA_{i,j}$, where $1 \leq i \leq m$ and $1 \leq j \leq n$, we calculate two different densities : *real data density* (or, shorter, data density) and *represented density*.

Data density is defined as $d_{i,j} = \frac{n_{i,j}}{A}$ where $n_{i,j}$ is the number of data points that fall into the sample area $A_{i,j}$. For a given visualization, the set of data densities is finite and discrete.

Represented density is defined as $rd_{i,j} = \frac{p_{i,j}}{A}$ where $p_{i,j}$ is the number of distinct active pixels that fall into $SA_{i,j}$. The number of different represented densities values a sample area can assume depend on its size. If we adopt sample areas of 8×8 pixels, as described before, the number of different not null represented densities is 64. Thus, we can represent at most 64 different represented density values. Because of collisions the following relation holds: $rd_{i,j} \leq d_{i,j}$.

Using the above definitions we devised an effective set of quality metrics whose complete discussion, however, is outside the scope of this paper (for more details, see³). Here we report a metric, CRSD, (Correctly Represented Sample area Differences) which is directly connected with our user study, i.e., a metric comparing the data density differences with the ones that are available on the final image.

The metric is based on the idea of comparing pairs of sample areas checking whether their relative data density (D) is correctly represented by their represented density (RD) (value that can be easily estimated exploiting the results shown in Figure 2). More precisely we check whether for any given pair of sample areas $SA_{i,j}$ and $SA_{k,l}$ one of the following rules holds:

$$\begin{aligned} \textit{if } D_{i,j} > D_{k,l} \textit{ then } RD_{i,j} > RD_{k,l} \\ \textit{if } D_{i,j} = D_{k,l} \textit{ then } RD_{i,j} = RD_{k,l} \\ \textit{if } D_{i,j} < D_{k,l} \textit{ then } RD_{i,j} < RD_{k,l} \end{aligned}$$

Introducing the $Diff(x, y)$ function defined as:

$$Diff(x, y) = \begin{cases} 1 & \text{if } x > y \\ 0 & \text{if } x = y \\ -1 & \text{if } x < y \end{cases}$$

we define the $match(i, j, k, l)$ function that returns true iff $Diff(D_{i,j}, D_{k,l}) = Diff(RD_{i,j}, RD_{k,l})$.

Moreover, in order to take into account the relevance of a comparison between two sample areas, we weight each comparison using the number of points falling in the two considered sample areas. Then, dividing the number of matching pairs by the weighted total number of pairs, we obtain a normalized weighted percentage of matching sample areas ranging between 0 and 1 (the greater the better). Roughly speaking we can say that, as an example, a CRSD value equal to 0.7 implies that 70 per cent of the data is represented preserving density differences.

The CRSD metric gives a general quality overview of the image and can be used, interactively, to make evident *where* are the most distorted areas; moreover, after having fixed a threshold value say $\Delta CRSD$, we can exploit it in two different automatic image optimizations:

1. given a data set and a prefixed display area size it is possible to check the quality metric against the threshold value and, in case of violation, to sample the original data set producing an image whose CRSD is greater than $\Delta CRSD$;

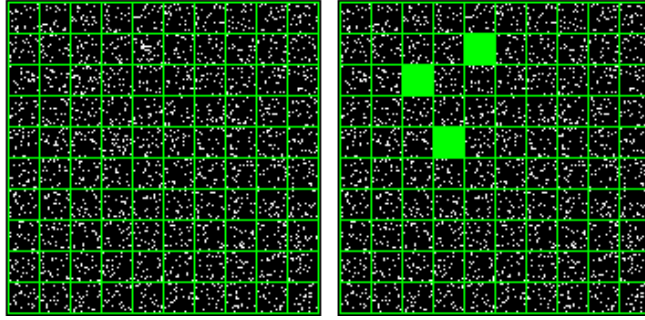


Figure 3. The areas filled with 25 pixels are not distinguishable from those filled with 38 pixels (those whose location is indicated with filled squares on the right side).

2. given a data set it is possible to compute the minimum display area size preserving the threshold value, allowing the system to optimize the screen usage.

The above metric, together with the statistical results, gives us the means to devise the automatic sampling technique described in the next section.

3.2. Optimal data sampling

Sampling is an effective technique to deal with cluttered 2D scatter plots; however, we need a guide to select the "optimal" amount of sampling. If the user is allowed for manually sampling the data, s/he is responsible to select the right factor and s/he does not have any indication about how her/his intervention improves the current image.

Exploiting the quality metric introduced above we can measure, for a given sampling factor, the quality of the generated image and then we can compute the optimum amount of sampling to apply. The idea is to find the sampling factor s that maximizes the CRSD metric.

Even if the problem is defined with continuous values of s , since very small differences in the amount of sampling do not produce perceptible visual differences, we devised an algorithm that runs with a fixed number of s values (i.e., 0.001, ..., 0.999, 1.000), obtaining a reasonable approximation of the optimal sampling factor.

This approach is not free of drawbacks. The problem is that the $Diff(x, y)$ function relies on the exact number of active pixels, and consequently, as an example, when comparing two sample areas with 50 and 49 active pixels (that look the same to the end user) the function returns the perceptually wrong information that the former is denser than the latter.

Figure 3 makes it clear. The image contains 100 sample areas, 97 of them sharing the same density (25 pixels) and 3 denser ones (38 pixels) highlighted in the picture (the areas at a higher density, in the left side of the figure, correspond the locations indicated by filled squares in right side). The difference is almost impossible to perceive and, consequently, it makes no sense to consider it.

These considerations convinced us to go into details on this aspect and to conduct the user study described in the next section.

4. THE USER STUDY

We devised a perceptual user study in order to answer a precise question: what is the minimum difference in active pixels between two sample areas that allows a user to perceive a density difference? In order to answer this question we asked the subjects to identify on a uniform background few denser areas.

We involved thirteen people in the user study (9 males and 4 females, ranging between 25 to 46) and, since the goal was to test exclusively perceptual issues, we did not classify them w.r.t. their skills and knowledge (we only asked the subject needing glasses to wear them). The involved people were undergraduate students, PhD students, and professors.



Figure 4. The user study main screen.

Basis/Delta	D1	RP1	D2	RP2	D3	RP3	D4	RP4	D5	RP5	DMIN
10	55	62	65	77	75	82	85	92	95	97	65
20	35	41	40	64	45	70	50	77	55	87	45
30	30	62	35	56	40	74	45	95	40	97	40
40	26	67	30	77	34	85	38	90	42	100	30
50	20	59	22.5	79	25	77	27.5	92	30	95	22.5
60	22	72	25	92	28	100	31	97	34	100	22
70	12	64	13.5	67	15	73	16.5	77	18	90	15
80	10	70	11.5	87	13	95	14.5	97	16	97	10
90	6	77	7	92	8	100	9	97	10	100	6

Figure 5. The user study results (all values are percentages).

According to the task, we presented the participants with the image shown on Figure 4. The image contains 100 sample areas, 97 of which filled randomly with the same number of active pixels (basis) while the remaining 3 are filled with extra pixels (δ). In the example the 97 sample areas are filled at 20% (basis=20) of their capacity and the 3 densest contain 150% more pixels than the basis ($\delta = 150$); in the figure the user identified and selected the uppermost densest sample area. A preliminary pilot study showed us that the δ value we were looking for depends on the basis and, in order to cope with this issue, we arranged the experiment as follows. All the users were presented (i.e., we performed a within subjects experiment) with 9 steps, corresponding to having the 97 equal sample areas progressively filled at 10%, ..., 90%. For each step, 5 substeps were performed, each of them showing 3 denser sample areas characterized by 5 increasing δ_i . The users were asked to select, for each substep, the 3 densest areas and the program recorded attempts and errors. We chose the 3 out of 100 percentage in order to have the tests completed in less than ten minutes, so avoiding to strain the subjects' eyes. Moreover, the pilot study allowed us to figure out reasonable values to assign to the δ_i values for each basis.

The results are collected in the table shown in Figure 5. The table shows per each row a different basis (10, ..., 90) and the five different increment steps (δ) adopted along the test (D1, ..., D5); for each increment the table shows the corresponding recognition percentage (RP1, ..., RP5) as well. As an example, the first row tells us that, while evaluating a basis of 10%, we asked the user to identify sample areas containing 55%, 65%, 75%, 85%, and 95% extra pixels and that the recognition rate was 62%, 77%, 82%, 92%, and 97%, respectively. We performed a single factor ANOVA test for each basis, confirming the significance of our figures (we obtained the worst result for the 50% basis: $F=4.79$, $F\text{-crit}=2.52$, and $P\text{-value}=0.002$). The last column shows, for each basis, the minimum increment we have to choose to guarantee that 70 out of 100 end users will perceive the density difference. Using such a column we can interpolate a function $minimum\delta(RD_{i,j})$ returning the minimum increment a sample area $SA_{k,l}$ must show to be perceived as denser than $SA_{i,j}$.

It is worth noting that such a function works only for represented densities because it uses increment percentages w.r.t. the pixels that belong to a sample area. If we consider, instead, data densities, we cannot compute the $minimum\delta(D_{i,j})$ function because data densities are computed in a pure metric space in which there is not any notion of pixel and resolution. In such a case, however, another restriction limiting the performance of our space can be considered: the human visual (point) acuity.¹² It is a well known fact that the human eye is not able to distinguish two points below the angle of 1'. This implies that, looking at a squared sample area of $l = 0,08$ inches from a typical distance of 20 inches, a human cannot distinguish two points closer than

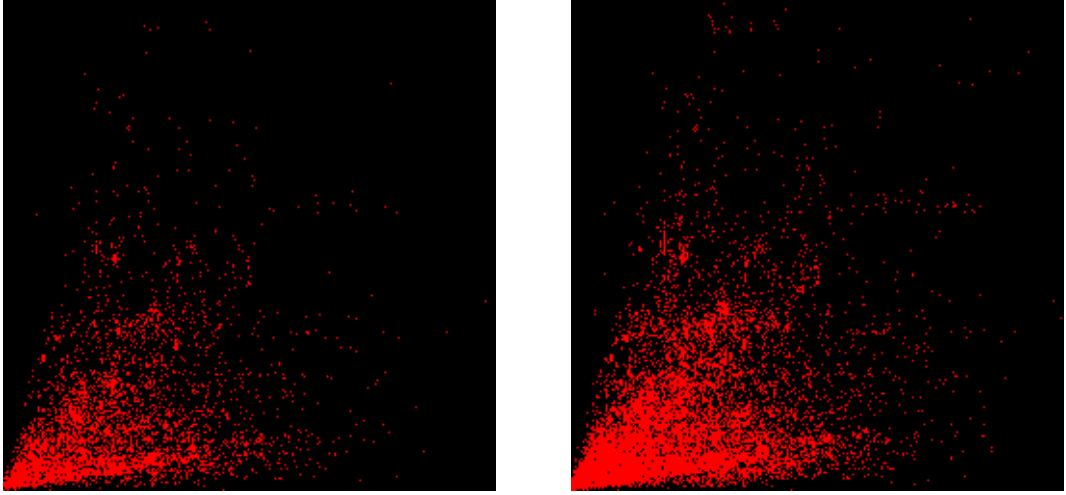


Figure 6. Optimum sampling ratio: (a)computed with Diff, (b)computed with PDiff.

$20 \times \tan(1') = 0,0057 \text{ inches}$. That corresponds to, approximately, an area of 14×14 virtual pixels. Under this assumption, we could extend the notion of collision to the metric space, being the collisions due to the human eye limitations. In this way we could reuse the statistical model described in Section 3 considering virtual pixels and extending the results to an hypothetical screen made of sample areas of $l = 0,08$ inches and containing 14×14 (172 DPI). By the way, this idea still needs further investigation and we will not report additional details about it in this paper.

The experiment results have been used to improve the accuracy of our algorithm, introducing the $PDiff(x, y)$ (Perceptual Diff) function as a modification of the $Diff(x, y)$ function introduced in Section 3.1:

$$PDiff(x, y) = \begin{cases} 1 & \text{if } x \geq y + y \times \text{minimum}\delta(y) \\ -1 & \text{if } y \geq x + x \times \text{minimum}\delta(x) \\ 0 & \text{otherwise} \end{cases}$$

In this way the quality metric deals with user perceptible vs numeric density differences. It is worth noting that the new $PDiff(x, y)$ function returns a lower number of differences than the $Diff(x, y)$ function, allowing the optimization algorithm to focus only on what really matters: the user point of view.

5. DISCUSSION

In this section we discuss the effect of the changes the user study induced on our formal environment.

Before that, we would like to discuss the user study results. The figures we obtained (Figure 5) are quite interesting and show, in our opinion, a non intuitive trend: the more a sample area is crowded, the lower the required δ to appreciate a density difference. The delta values range from 65% of increment, for very sparse areas (10% used area), to 6% for very crowded areas (90% used area).

As regards our environment, we tuned our algorithms using the new $PDiff(x, y)$ function that affects both the overall CRSD quality metric and the calculation of the optimal sample ratio. We applied our algorithms against the postal parcel visualization. The image comes from a tool specifically developed for our purposes whose functionalities can be seen in Figure 7. It is a Java based application that permits to inspect several characteristics of the displayed image such as: the data density and the represented density of each sample area, a zoomed representation of the selected sample area showing where pixels overlap, a histogram of data densities and represented densities, some quality metrics, and the number of overlapping pixels. It is also possible to compute and apply the optimal data sampling using both the $Diff(x, y)$ and $PDiff(x, y)$ functions.

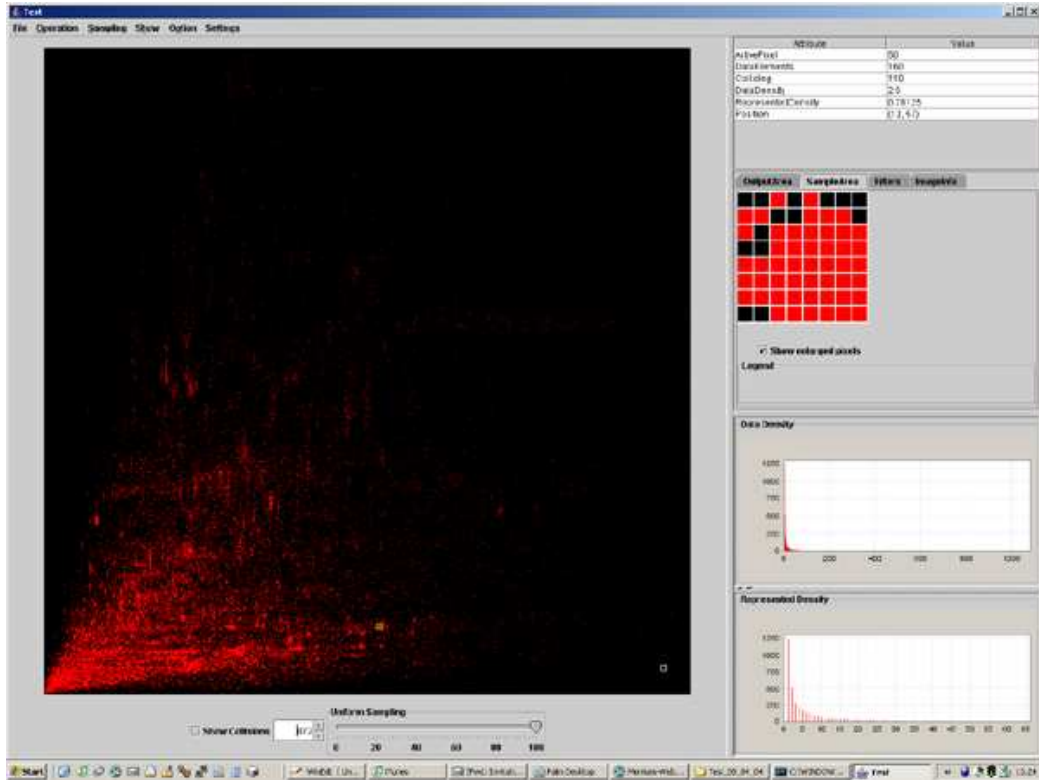


Figure 7. The original data set.

Figure 6 shows the application of different values of sampling suggested by applying the $Diff(x, y)$ function and the $PDiff(X, Y)$ function, respectively. The original image (with no sampling) presents a CRSD value of 0.89 and, using the $Diff(x, y)$ function, an optimal sample ratio of 12% (88% of the data has been removed) is suggested, producing a new CRSD equal to 0.95. The $Diff(x, y)$ function deals with 58,591 numerical data density differences. On Figure 6 (b), we show the result coming from the $PDiff(x, y)$ function. In this case, the algorithm detects 46,096 perceptual differences, 22% less than before, that is the number of differences that can really be perceived. Having less constraints to satisfy, the new algorithm produces a more effective optimal factor of 23% shown in the picture. An advantage of this new value is that the very sparse areas (right uppermost points) are still visible in the picture while the 12% sampling factor quite destroys them. Moreover, the user has a numeric feedback about how well the current image represents what is inside data or, in other words, s/he has an indication about to what extent s/he can trust what s/he sees.

6. CONCLUSIONS AND FUTURE WORK

In this paper we presented a user study aiming at understanding the way in which differences in densities are perceived by the users. The results we obtained from this study have been used to tune a formal framework able to preserve in an automatic way relative densities. Such an integration showed that a better understanding of human perception of density can improve the image optimization processing.

Several open issues rise from this work:

- the issue of visual perception deserves a deeper analysis. Even if the results are interesting and improved our understandings, during the user study we had a lot of intuitions about how to extend our framework. First of all, we discovered that several more factors can influence the users' perception, like the sample area dimension and the presence of patterns. Moreover some issues, intentionally not covered by our study, like colors, saturation, contrast, could be included in the test.

- quality metrics. We tackled the relationship existing between device limitation (i.e., pixels) and human eye limitation (i.e., visual acuity) but the overall picture is not totally clear. It is our intention to study the matter in depth, obtaining an overall quality metric framework, considering all the involved parameters and limitations.

We are actually extending the prototype functionalities to apply and verify our ideas. We want to implement a dataset generator to conduct controlled tests. The dataset generator will permit to generate artificial distributions, giving the possibility to control specific parameters, that will be used to create specific cases considered critical or interesting.

ACKNOWLEDGEMENTS

We would like to thank Pasquale Di Tucci for his invaluable help in implementing the software prototype. Work supported by the MIUR-FIRB project "MAIS" (Multichannel adaptive Information Systems, <http://www.mais-project.it/>)

REFERENCES

1. M. Trutschl, G. Grinstein, and U. Cvek, "Intelligently resolving point occlusion," in *Proceedings of the IEEE Symposium on Information Visualization 2003*, p. 17, IEEE Computer Society, 2003.
2. G. Ellis and A. Dix, "Density control through random sampling: an architectural perspective," in *Proceedings of Conference on Information Visualisation*, pp. 82–90, July 2002.
3. E. Bertini and G. Santucci, "Quality metrics for 2d scatterplot graphics: automatically reducing visual clutter," in *Proceedings of 4th International Symposium on SmartGraphics*, May 2004.
4. N. Miller, B. Hetzler, G. Nakamura, and P. Whitney, "The need for metrics in visual information analysis," in *Proceedings of the 1997 workshop on New paradigms in information visualization and manipulation*, pp. 24–28, ACM Press, 1997.
5. E. R. Tufte, *The visual display of quantitative information*, Graphics Press, 1986.
6. B. Richard, "Concept demonstration: Metrics for effective information visualization," in *Proceedings For IEEE Symposium On Information Visualization*, pp. 108–111, IEEE Service Center, Phoenix, AZ, 1997.
7. G. G. Grinstein, P. E. Hoffman, and R. M. Pickett, "Benchmark development for the evaluation of visualization for data mining," *Information visualization in data mining and knowledge discovery*, pp. 129–176, 2002.
8. H. Levkowitz and G. T. Herman, "Color scales for image data," *IEEE Computer Graphics and Applications* **12**(1), pp. 72–80, 1992.
9. L. D. Bergman, B. E. Rogowitz, and L. A. Treinish, "A rule-based tool for assisting colormap selection," in *Proceedings of the 6th conference on Visualization '95*, p. 118, IEEE Computer Society, 1995.
10. C. G. Healey and J. T. Enns, "Large datasets at a glance: Combining textures and colors in scientific visualization," *IEEE Transactions on Visualization and Computer Graphics* **5**(2), pp. 145–167, 1999.
11. R. Kosara, S. Miksch, H. Hauser, J. Schrammel, V. Giller, and M. Tscheligi, "Useful properties of semantic depth of field for better f+c visualization," in *Proceedings of the symposium on Data Visualisation 2002*, pp. 205–210, Eurographics Association, 2002.
12. C. Ware, *Information visualization: perception for design*, Morgan Kaufmann Publishers Inc., 2000.