Incorporating Visualization Research in Introductory Programming Course: Case Studies

Sunghee Kim

Department of Computer Science, Gettysburg College, U.S.A

Abstract

The importance of early research experience for undergraduate students has been stressed time and time again. This paper presents three case studies in which non-CS major students could gain a visualization research experience in their first programming course. In all case studies, students were given real climate data to visualize. In the first case study, students visualized spatial correlation between two variables (weather conditions) on a map so that viewers could infer areas in which the two variables were highly correlated in a positive or negative way, or areas with little to no correlation. In the second and third case studies, students generated single variable visualization and multidimensional visualization of two or four variables. In each of the three case studies the students were led through the process of understanding data, exploring different representations, and designing and implementing an agreed-upon visual representation. Increased number of students decided to take the next course in Computer Science compared to previous years without a research project. Feedback from the students suggests that they enjoyed using data they could understand and found the process and the final product rewarding and applicable to projects in their major and courses.

CCS Concepts

•***Human-centered computing*** → *Visualization;*

1. Introduction

There has been an increasing focus on undergraduate research, especially in the STEM fields. Exposing students to research very early in their undergraduate curriculum may help them stay in the STEM majors and motivate them to go to graduate school. Undergraduate research participants get an idea of what a career in science would be like and are better prepared for delving into research in graduate school than those who have not participated in research and learned to be independent [Web07, Lop07].

In general undergraduate students in our department are involved in research after they have completed Data Structures. We have been discussing ways to involve students in research in their first or second year and we believe that we found a vehicle in a recently developed non-major course described in Section 2.

2. Course Description

The course, Introduction to Scientific Computing, introduces students to the fundamental principles of computer programming, algorithmic thinking, and problem-solving with particular emphasis on applications in the sciences. Most lecture examples and assignment problems are drawn from the STEM fields such as mathematics, biology, chemistry, physics, and health sciences. Students also write solutions to problems in economics, statistics, and create simple visualizations of data such as scatter plots, histograms, and bar charts.

The course targets mainly current and future mathematics or science majors as well as those who intend to major in Mathematical Economics or Psychology. This course is considered equivalent to CS1 and serves as an alternate entry to the CS major, i.e., students can enroll in CS2 upon successful completion of this course.

After discussion with faculty in other science departments, we chose Matlab as the formal programming language for this course over Java which we use in our regular CS1 course. Matlab offers advantages such as the ability to easily handle a wide variety of common and domain-specific data formats, quick testing of algorithms without recompilation, very little setup for writing functions and scripts, and type-less data, to name just a few. The learning is expected to be faster than other high-level languages.

For those who wish to continue in Computer Science after this course, we include three to four weeks of "transition to Java" at the end of the semester.

3. Visualization Research Project Case Studies

3.1. Motivation

Computer-generated images are playing increasingly bigger roles in every facet of our life. Computer imaging provides visually stunning art for games and movies, aids surgeons in planning and performing surgery, and ensures privacy and security using biometrics. These are only a few samples of computer imaging in our life.

"A picture is worth a thousand words." One of the main goals in visualization research is to provide insight by presenting data in such a way that human observers can infer not just apparent pattern in the data but also explore and discover unknown patterns as well [MDB87]. A well-designed visual representation of data can readily give us information which users may not have understood if data were presented in words or in textual format only.

Visual computing is believed to be a good way to engage and arrest student interest in programming. Students can see the result of their programming immediately and in a visually appealing way. They can start imagining what else is possible when they see what they can do with very little programming experience. According to research, visual computing is easy to use, easy to learn, and improves productivity [WB97].

3.2. Custom Graphics Library

Although Matlab offers full 2D and 3D graphics functionalities, it is difficult to use for the students who are new to computer programming.

We provide a basic graphics library with the following functions:

- `Canvas`: creates a Canvas of custom dimensions
- `drawText`: draws text on the Canvas at specified position in selected color and font sizes
- `drawPoint`: draws a point
- `drawLine`: draws a line
- `drawCircle`: draws a circle outline
- `drawOval`: draws an ellipse outline
- `drawRect`: draws a rectangle outline
- `drawTri`: draws a triangle outline
- `drawPolygon`: draws a closed polygon outline
- `drawImage`: draws a specified image

For the closed shapes, e.g., circle, oval, rectangle, triangle, and polygon, `fillXXXX(..)` is used where `XXXX` is replaced by `Circle`, `Oval`, `Rect`, `Tri`, and `Polygon` respectively.

In addition to the standard command-line input provided by Matlab, support is provided for processing a single or multiple mouse clicks.

3.3. Dataset

The data used in all case studies is the Climatic Research Unit global climate dataset which consists of a multidimensional 0.5 degree latitude by 0.5 degree longitude resolution monthly averages of eleven weather conditions collected for positive elevations throughout the world from 1961 to 1990 and averaged over these 30 years by the Intergovernmental Panel on Climate Change (IPCC Data Distribution Center. http://www.ipcc-data.org).

Of the eleven conditions collected, we used the four conditions that were most familiar to the students: temperature, vapor pressure, precipitation, and wind speed.

Data was further divided into regions of dimensions 122 x 61. In all case studies, the visualization of this data included scaling up by a factor in both directions.

Parts of the world in which no data values existed were marked -9999. The non-value entry had to be specially handled in the case studies in both pre-processing computation and visualization.

4. Case Study 1: Spatial Correlation Visualization

For the first case study, the goal of the project was to display the correlation between two variables. In our first attempt to introduce data analysis and visualization as a project, there were strict guidelines how the students should represent the data.

After being introduced to statistical measure of association between two independent variables and applications, students were expected to submit their work in three parts. They were encouraged to work in pairs with another student of a similar level. A pair could be formed after the instructor ensured that the students' grades on the first exam and assignments were within one letter grade; e.g., students could work together if their grades were A-B, B-C, or C-D, but not A-C or A-D.

4.1. Part I Correlation of One-dimensional Variables

In part one, the students derived and wrote functions necessary to compute the correlation coefficient of two one-dimensional variables. This included calculating z-score, covariance, and correlation coefficient between two equal-length vectors.

4.2. Part II Correlation of Two-dimensional Variables

In part two, the students extended the one-dimensional correlation from part one to two-dimensions without using the built-in Matlab functions in order to gain more programming experience.

Up to part two, the students used small artificial data for testing to verify the correctness of their functions and compared their results with built-in correlation functions.

4.3. Part III Correlation Visualization

In part three, the students were to consider the properties of the climate data and carefully exclude non-value locations. They compiled a full correlation map between two variables and generated a visualization of the correlation map using two colors. We used a diverging color scheme (blue to red) which is considered useful for subjective interpretation of progression from a critical midpoint of the data range, in our case 0 (no correlation) to blue (negative correlation) and red (positive correlation).

See Figures 1 and 2 for examples of student submissions visualizing correlation maps.

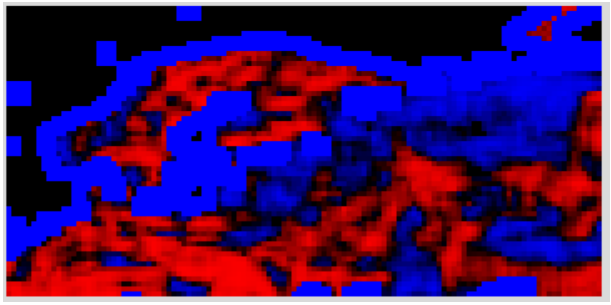

Figure 1: *Correlation map between vapor pressure and precipitation in Scanadinavia Region.*

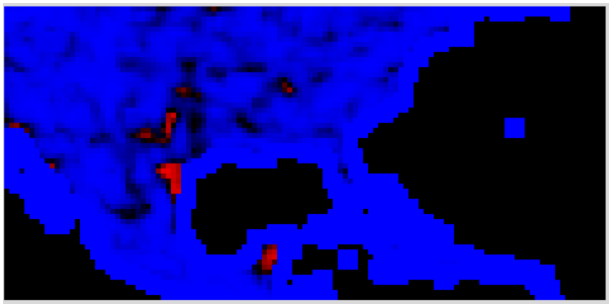

Figure 2: *Correlation map between vapor pressure and temperature in the U.S. and Central America Region. Highly negative correlation can be observed in most of the region.*

4.4. Class Discussion

Students were introduced to several applications of the correlation coefficients in various fields. For example, with census data we could find how education level is correlated with income level. With climate data we could infer the relationship between vapor pressure and precipitation. With healthcare data we could infer how soda consumption is related with obesity. With stock market data we could examine oil prices and performance of certain stocks. Students were asked to think about other examples in which correlation might exist and how they would formulate hypotheses and collect data for their own projects in their non-computing courses.

Another topic of discussion was on the limitations of the chosen representation. In this simple representation, when users see a black area in the map it is impossible to differentiate non-value, i.e., undefined, and zero-correlation areas. This would be particularly problematic if the viewers are not domain experts. The zero-correlation areas also make the visualization less visually appealing. This was left as an open question so that the students could reflect more deeply and independently about better representations or possible improvements of the current method.

5. Case Study 2: Multidimensional Visualization 1

Building upon the success of Case Study 1, in the following semester we made the project more visualization-focused. An open problem in visualization is how to effectively visualize multiple variables in a single image. In many fields, it is important for domain experts to understand not just individual values of a single variable but also the relationship between multiple variables. There still is not a clear and conclusive guideline for multidimensional visualization despite active research into this problem for more than two decades.

For the second case study, the students were expected to visually represent a single variable, two variables, and four variables in a single display. For single variable visualization, they were to implement several visual representations in grey scale or color: filled rectangle, cross marks, horizontal line textures, and vertical line textures. For multi variables, they were to use the co-presentation method described in [HSKTH07] and [Mil07] which divides a data area by the number of variables and each divided region represents one variable.

5.1. Process

The students were introduced to the topic of multidimensional visualization including a brief presentation of recent research results. Similar to Case Study 1, the students were expected to submit their work in three parts over a month and half and were encouraged to work with a partner of similar level.

5.2. Part I Visual Mappings

In part one, the students were asked to write Matlab functions to visually represent the value of a single variable in a specified rectangular data region. The visual markers were: grey intensity, color intensity, cross, and horizontal or vertical lines (see Figures 3 and 4). These markers were selected not only for their simplicity in understanding and implementation, but also for the opportunity to demonstrate the effect of different markers on the quality of the visualization.

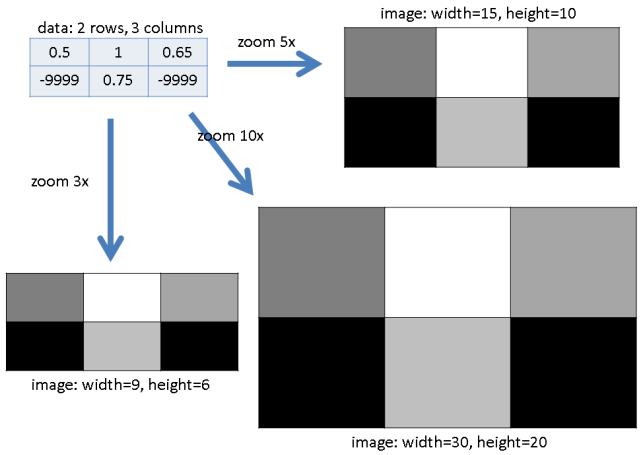

Figure 3: *Example of Single Variable Visualization in Grey Scale Image in three scales. Test data is 2 x 3 with two undefined values.*

The following functions visualized one data point in a given color at a specified region.

- `colorDataPoint(x, y, w, h, color)`
- `crossDataPoint(x, y, w, h, color)`
- `horLineDataPoint(x, y, w, h, color)`
- `verLineDataPoint(x, y, w, h, color)`

As part of matrix (2D array) processing, the students wrote code to compute maximum and minimum values of the data, ignoring undefined values (-9999). The data was then linearly scaled to fit the whole range of a color channel. All the functions students wrote in this part were then used to visualize 1, 2, or 4 variables of real data.

- `maxMatrix(data)`
- `minMatrix(data)`
- `linearScaleMatrix(data, min, max)`

5.3. Part II Single Variable Visualization

In part two, the students wrote functions to visualize 2D data using a particular visual mapping:

- `drawScaledGray(data, sf)`
- `drawScaledColor(data, sf, color_channel)`
- `drawScaledHor(data, sf)`
- `drawScaledVer(data, sf)`

The test data used in this case study was the same climate data from the first case study and was 122 x 61. A scaling factor, sf=100, was used to create a larger image as seen in Figure 4 for artificial 2 x 3 data. The color_channel parameter for `drawScaledColor` was 1 for red, 2 for green, and 3 for blue.

For both grey and color mappings, high intensity (bright color) represented a high value while lower intensity (darker color) represented a low value for the variable (Figure 4 top and middle rows)

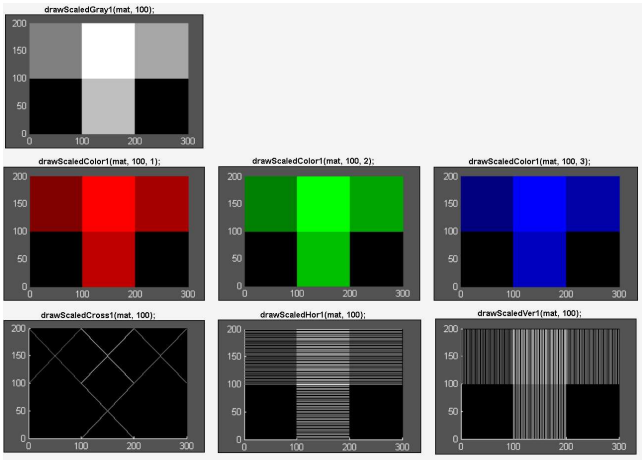

Figure 4: *Visual Representation of Single Variable. Top row: grey scale intensity; Middle row: color intensity; Bottom row: crosses, horizontal lines, and vertical lines.*

When horizontal or vertical lines were mapped to represent variables, the density of the lines represented high or low value for the variables. For example, in a rectangular data area, more lines indicated higher value (Figure 4 bottom row, center and right).

With the cross markers, the intensity of the markers was mapped to the value (Figure 4 bottom row, left). The students noted that the crosses were least helpful in perceiving data among all visual markers they tried.

An example of a single variable visualization is shown in Figure 5.

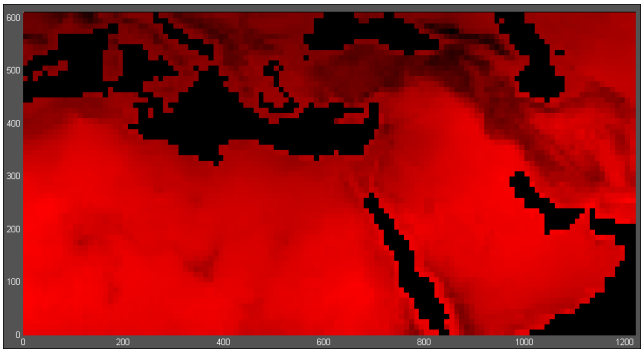

Figure 5: *Example of Single Variable Visuzation of Temperature in North Africa and Middle East Region. Bright red region had higher average temperature than darker areas.*

5.4. Part III Multidimensional Visualization

In part three, the students generated images for visualizing two variables or four variables simultaneously. With the minimum scaling factor of 4, each data region (representing one location in data) was split into four equal-sized sub-regions.

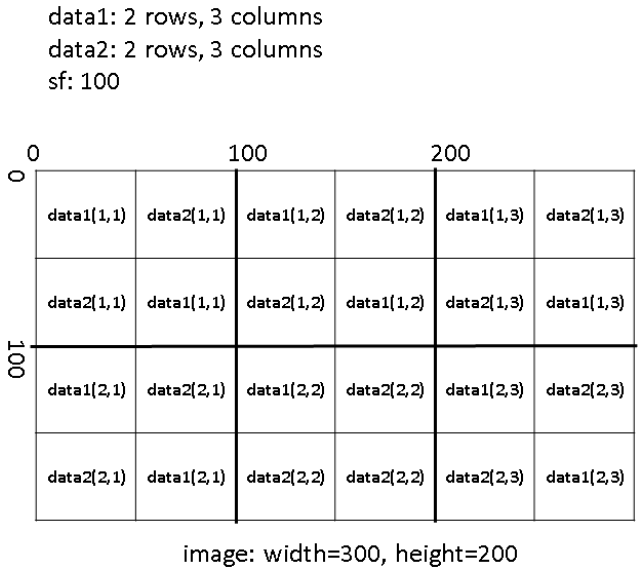

Figure 6: *Sharing of Screen Space for Two Variables.*

For the visualization of two variables in the same region, top-left and bottom-right sub-regions represented the first variable, while

top-right and bottom-left sub-regions represented the second variable (Figure 6). Examples of two-variable visualizations of a 2 x 3 artificial dataset are shown in Figure 7.

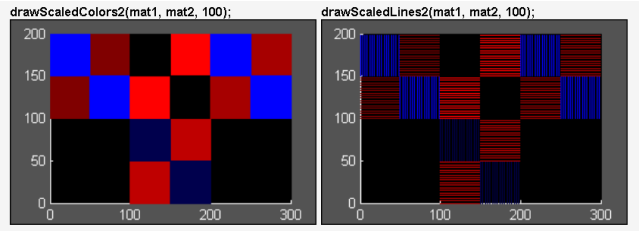

Figure 7: *Visualization of Two Variables. Left: blue and red colors for two variables. Right: Blue and vertical lines for the first variable, Red and horizontal lines for second variable.*

In Figure 8, the temperature and vapor pressure in North Africa and Middle East Region are visualized using two colors only. In this example, temperature is represented by red color and vapor pressure by blue color.

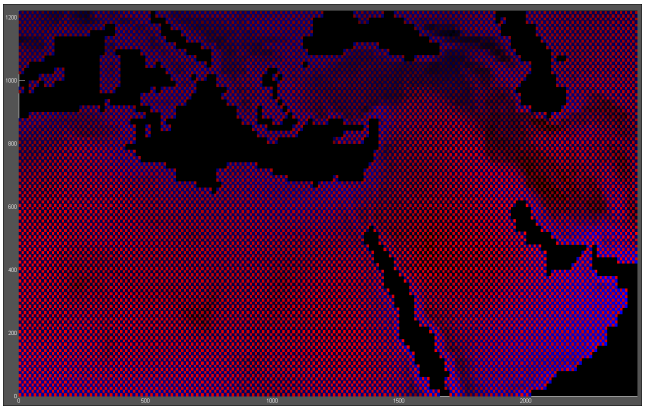

Figure 8: *Visualization of Temperature and Vapor Pressure in North Africa and Middle East Region.*

For the visualization of four variables the screen space is split into the same four equal-area/shape sub-regions and then each of the four variables is exclusively visualized in one sub-region. It is somewhat difficult to fully appreciate the figures included in this paper since they are much smaller than their full sizes.

Students wrote the following functions for this part:

- `drawScaledColor2`: two colors
- `drawScaledLines2`: horizontal, vertical
- `drawScaledColor4`: four colors
- `visSingle(data, sf)`: single variable vis.
- `visMulti2(data1, data2, sf)`: two-variable vis.
- `visMulti4(data1, data2, data3, data4, sf)`: four-variable vis.

For the `visSingle` and `visMulti` functions, the students were asked to experiment with different visual mappings and use the combination of mappings that were most helpful in communicating the data.

As the final step, the students added a simple user interface for the user to choose the number of variables to visualize and specify the corresponding data files. Their programs perform file processing, data pre-processing, and visualization seamlessly.

5.5. Class Discussion

Several important questions were raised in class discussion. Does the visualization display the values of each variable clearly? Is the relationship between two or four variables visible? How should odd number of variables be handled? These are open questions that could lead to further research.

6. Case Study 3: Multidimensional Visualization + Chernoff Face Study

Following the projects in Case Studies 1 and 2, we modified the project to include more research component and direct feedback. In addition to color and lines, students wrote a function to visualize a value by drawing a filled ellipse in the given color at the center of a specified region. The size and color of the ellipses was used to encode two variables.

In part 2 of the project, students chose a geographic region (Scandinavia/North-Central Europ/Western Russia, Russia, North Africa/Middle East, China/India/Central Asia, Central Africa, US/Central America) and generated two-varible visualizations using the following 7 different combinations:

- `color, vertical lines`
- `color, horizontal lines`
- `vertical lines, horizontal lines`
- `color, color of ellipses`
- `size of ellipse, lines`
- `color of ellipses, lines`
- `color and size of ellipses`

An example visualization of two variables is shown in Figure 10.

Students were then asked to examine whether (1) the value of each variable can be read and (2) the relationship, if any, between two variables can be understood and provide written observations of each visualization.

The students reported that combining lines with ellipses did not work very well as the two visual markers tended to cover each other (ellipse covering lines or lines making ellipse difficult to see). The lines in general were not helpful and made the final visualization too cluttered. They also reported that using color and size of ellipses worked well as they could identify where and how the color and/or sizes of the ellipses change. Combination of two different colors (color for the first variable, color of equal-size/shape ellipses for the second variable) received mixed reviews from the students. Some students reported that the colors made it easy to see both how a variable changed and also how two variables seemed to affect each other. Others were not satisfied with the results because they found it difficult to identify two colors that were easy to distinguish.

In part 3, the students read the paper on Chernoff faces [Che73] and discussed in writing the advantages and disadvantages of using the Chernoff faces, and whether and how Chernoff faces should be

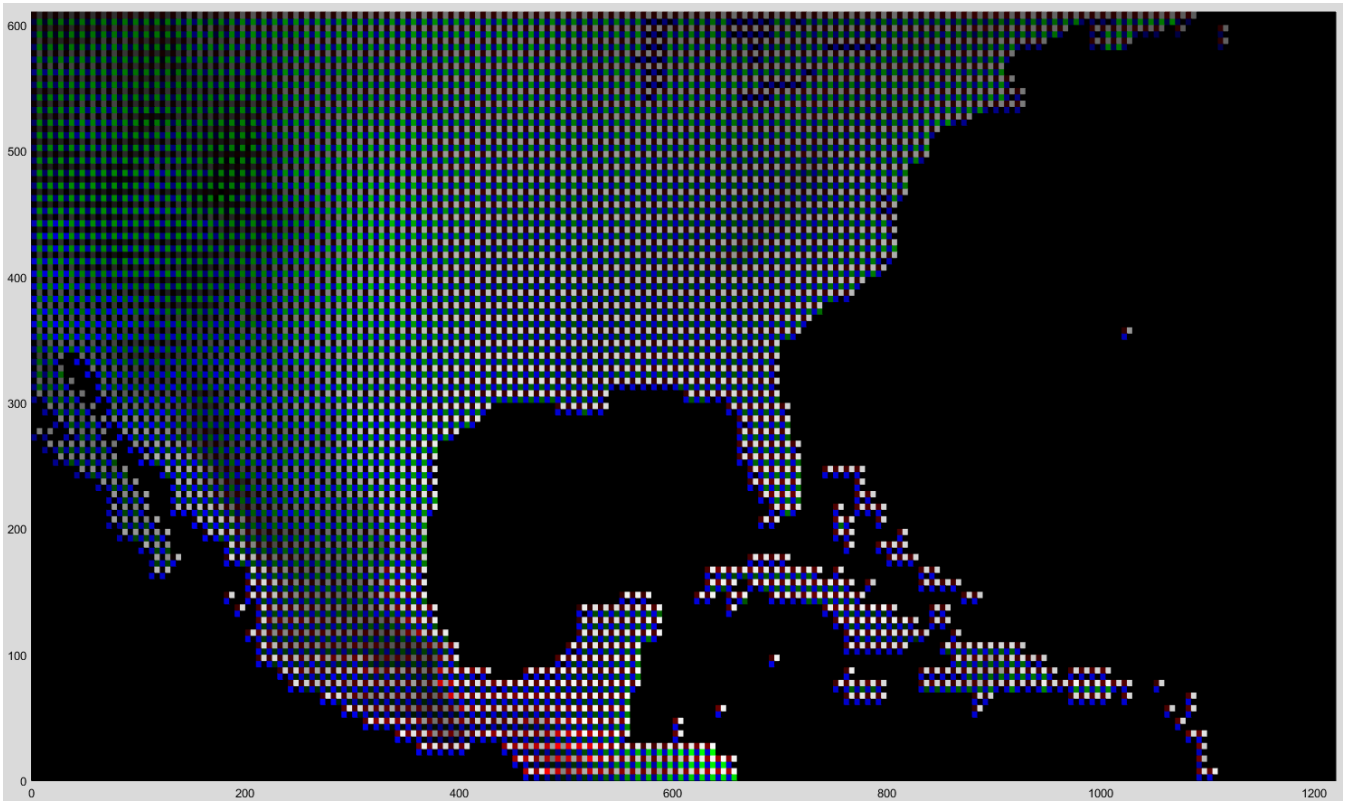

Figure 9: *Visualization of Temperature, Vapor Pressure, Wind Speed, and Precipitation in the U.S. and Central America Region.*

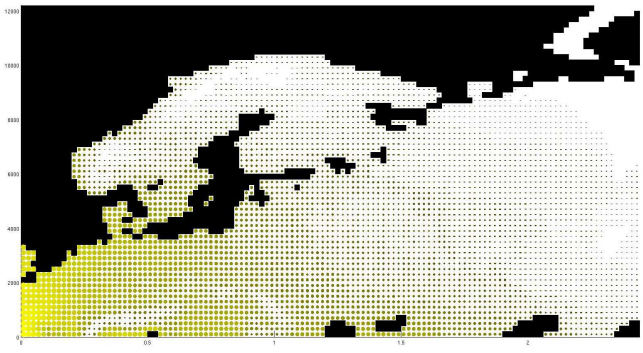

Figure 10: *Student Submission Using Color and Size of Ellipses.*

used to visualize dense spatial data such as the climate data used in the Case Studies.

They also evaluated the visualization project component of the course by completing a survey that included 3 questions rated on 1 to 5 scale and 2 questions for qualitative feedback. The ranges of responses from the quantitative portion of the survey were: (Q1) Level of Difficulty 3 ~4 (average 3.25); (Q2) Level of Personal Interest 2 ~5 (average 3.7); and (Q3) Level of Relevance to Chosen Major 1 ~5 (average 3.6).

Figure 11 shows sample responses to the discussion questions: Q4. Discuss how similar visualization can be helpful in your major; and Q5. free-form comments.

7. Discussion

Our goals of including visualization research projects in the introductory programming course are two-fold. The main goal is to expose the students majoring in other fields to the power of computing and visualization, and its applications in their chosen fields of study. The other goal is to attract more majors/minors. The latter goal is particularly challenging since this course is primarily for students who are already majoring or who intend to major in other sciences and related disciplines.

7.1. Enrollment Study

Our classes have an enrollment cap of 16 with 8 seats reserved for first-year students. Although this makes it difficult to reach a concrete conclusion from three small classes, we did draw more students to Computer Science courses and major/minor than previous semesters without a visualization project.

Specifically, after the semester with Case Study 1, four out of thirteen (30.7%) students enrolled in CS2. All four were first-year students. One declared CS major immediately while the other three

Q4. Applications in their major project/courses
"very frequently graphs and different visualizations are used to demonstrate economic facts. An example application could be visualizing the differences in GDP per capita, unemployment rate, household size, and total GDP per region across the country."
"We're working with data and analyzing numbers which directly is related."
"What we learned in this project could be applied to data analysis."
"Looking at data from different areas is important to my major. This project taught me how to display data properly so it is not misleading."
"I really appreciated this proejct because it is very closely related to many courses I have taken in biology such as analyzing migration patterns and population distributions. This project could almost be taken as-is. I also enjoyed using programming with the intention of visualizing variables in a useful manner."
"Math & Physics need representation of variables. This can find applications both in quantum mechanics and in relativity where we introduce a fourth dimension."
"When we are trying to represent the relationship of different variables that affect the economy, what we did in this project is really useful. For example, we can generate a graph to see how the populations, levels of educations, and ages relate to the GDP of different regions."
other comments
"facscinated by how the addition of color in different forms and magnitude changed how I saw an image"
"this project definitely made me think about how to better visualize data. I also realized how difficult it is to make an image with multiple variables on it recognizable."
"never carefully considered how data was displayed before ... now understand why data is displayed in certain ways and that being able to tell the difference between variables is important"
"this has made us really think about how to incorporate multiple data inputs in one graphic."
"did make us think about how to visualize data. The project was very adequate to the level of the class but at the same time gave us an occasion to improve our skills."

Figure 11: *Qualitative Feedback.*

were undecided whether to pursue a major or minor in Computer Science.

After the semester with Case Study 2, four out of twelve (33.3%) students enrolled in CS2. Three were first-year students who were not initially intending to major in CS but two of them eventually decided to become CS majors. One sophomore majoring in Mathematics also enrolled in CS2 intending to minor in Computer Science.

After the semester with Case Study 3, three out of twelve (25%) students enrolled in CS2. One of these three students was a Physics major who decided to minor in Computer Science.

For enrollment comparison with three previous semesters taught by the same instructor without a visualization project, one out of thirteen (7.6%), three out of twenty seven (11.1%, 2 sections), and two out of thirteen (15.3%) students enrolled in CS2 after their respective semesters.

It is difficult to definitively attribute the increase in the enrollment in CS2 to the visualization projects. However, from informal discussions with students, we believe that the visualization projects had a positive impact on the perception of Computer Science. One student wrote, "The data visualization is really really cool." Another student wrote, "I liked seeing all the work I put in come together to create such a cool graphics map."

7.2. Challenges

There were challenges in including a directed research project in the course. The main challenge we faced was that when research was assigned as a part of course work, some students tended to consider it just another assignment to get over with and therefore did not seem to enjoy or be fully engaged in the creative process. They only seemed to be relieved when the final visualization was completed and submitted. One solution to this challenge may be to allow students to find and use their own data as long as it meets certain requirements.

Another challenge was that since the students were in their first programming course, they could not start the implementation phase of the project until a month or a month and a half into the semester. This meant that they had to complete a significant portion of the work near the end of the semester further leading to "less creativity." It is important to design the course in such a way that the students can be engaged in the research project throughout the whole semester. Perhaps the students could be involved in the project by starting on the data transformation phase of the visualization pipeline process [CMS99] as early as when they learn basic expressions and functions. Students then can reach project milestones in line with the basic programming concepts they learn throughout the semester.

8. Future Work

In future course offerings, we would like to explore other multidimensional spatial data visualization algorithms. For example, instead of presenting the values of different variables side by side, they could be layered similar to the method described in [BHW06]. Another algorithm to consider is to use multiple visual representations such as color, density, size, orientation, and texture of an icon or glyph to represent different fields as discussed in [WH01, Gah98, CLKH14].

Another extension is to include an additional research component such as running a small user study to evaluate the implemented

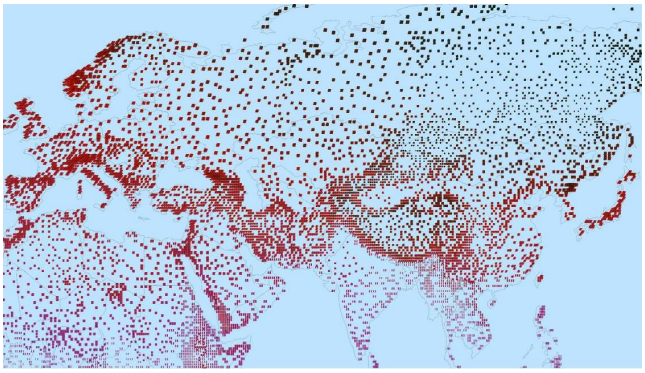

Figure 12: *Using Glyph Attributes for Visualizing Temperature, Vapor Pressure, Wind Speed, and Precipitation during January over Europe and Asia. [WH01]*

visualization algorithm or a comparative user study between different types of multivariate spatial data visualization. Including a user study as part of the course without comprosmising other core contents will be quite challenging due to the time and curricular constraints. However, if implemented, the experience of desining and conducting a user study to evaluate the effectiveness of a new algorithm will introduce the students to a very important aspect in data visualization research [KHI*03, ML17].

9. Conclusion

In this paper, we presented three case studies in which visualization research was a large part of an introductory programming course. Using climate data, which was familiar and easy to understand, students computed the statistical properties of the data and generated visualizations of single and multiple variables, and of the relationship between two variables. Through the design and implementation process and class discussions students gained experience with understanding data and visualizing some of its properties. We believe that this is a way to provide studentw with early exposure to research in Computer Science and applications of visualization in other fields. Written feedback indicated that students had a positive and rewarding experience.

References

[BHW06] BAIR A., HOUSE D., WARE C.: Texturing of layered surfaces for optimal viewing. *Transactions on Visualization and Computer Graphics 12*, 5 (Sept. 2006), 1125–1132. 7

[Che73] CHERNOFF H.: The use of faces to represent points in k-dimensional space graphically. *Journal of the American Statistical Association 68*, 342 (June 1973), 361–368. 5

[CLKH14] CHUNG D., LARAMEE R., KEHRER J., HAUSER H.: *Glyph-based Multi-field Visualization*. Springer, 2014. C.D. Hansen and M. Chen and C.R. Johnson and A.E. Kaufman and H. Hagen. 7

[CMS99] CARD S., MACKINLAY J., SHNEIDERMAN B.: *Readings in Information Visualization: Using Vision to Think*. Morgan Kaufmann, 1999. 7

[Gah98] GAHEGAN M.: Visualization techniques for exploratory spatial analysis. *Computers, Environment and Urban Systems 1* (1998), 43–56. 7

[HSKTH07] HAGH-SHENAS H., KIM S., TATEOSIAN L., HEALEY C.: Weaving vs. blending: A quantitative assessment of the information carrying capacities of two alternative methods for conveying multivariate data with color. *Transactions on Visualization and Computer Graphics 13*, 6 (Nov. 2007), 1270–1279. 3

[KHI*03] KOSARA R., HEALEY C., INTERRANTE V., LAIDLAW D., WARE C.: Thoughts on user studies: Why, how, and when. *Computer Graphics and Applications 23*, 4 (2003), 20–25. 8

[Lop07] LOPATTO D.: Undergraduate research experiences support science career decisions and active learning. *CBE Life Science Education 6*, 4 (2007), 297–306. 1

[MDB87] MCCORMICK B., DEFANTI T., BROWN M.: Definition of visualization. *Siggraph 21*, 6 (Nov. 1987), 3. 2

[Mil07] MILLER J.: Attribute blocks: Visualizing multiple continuously defined attributes. *Computer Graphics and Applications 27*, 3 (May 2007), 57–69. 3

[ML17] MCNABB L., LARAMEE R.: Survey of surveys (sos) - mapping the landscape of survey papers in information visualization. *Computer Graphics Forum 36*, 3 (June 2017), 589–617. 8

[WB97] WHITLEY K., BLACKWELL A.: Visual programming: The outlook from academia and industry. In *Proc. Workshop on Empirical Studies of Programmers '97* (1997), pp. 180–208. 2

[Web07] WEBB S.: The importance of undergraduate research. *Science* (July 2007). 1

[WH01] WALTER J., HEALEY C.: Attribute preserving dataset simplification. In *Proc. Visualization '01* (2001), pp. 113–120. 7, 8