Creating a Comprehensive Pipeline for Exploring Cultural Variance in Data Visualization

Major Qualifying Project

Written By:

Joselin Barbosa

Katie Bowles

Vivian Reno

Advisors:

Lane Harrison

Noëlle Rakotondravony

A Major Qualifying Project

WORCESTER POLYTECHNIC INSTITUTE

Acknowledgements

The team would like to thank our advisors Professor Lane Harrison and Noëlle Rakotondravony for their dedication to this project. Without their full support and guidance this project would not be what it is.

Abstract

Executive Summary

Introduction

To explore the differences in data visualizations across diverse cultural landscapes and analyze trends in the colors used, we implemented a process and set of tools for the collection, storage, processing, and display of visualizations and their color data trends. Our pipeline allows for growth and customization; it consists of a process for collecting data visualizations from websites and social media platforms, a palletizer program for extracting images' raw color data, and a website to display the visualizations and color data for analysis. We also demonstrate our pipeline by applying it to a sampling of data visualizations from Ecuador, Indonesia, and Nigeria.

Background

Data visualizations are used across many types of media, including social media platforms, newspapers (digital and print), and magazines. Along with lending the medium legitimacy in the eyes of uneducated viewers, data visualizations allow all viewers both to verify that the conclusions of the work being supported by the visualization are actually grounded in the data, and to discover trends and patterns in the data that are not otherwise presented. The impact of data visualizations varies based on intent, but this impact is often greater than that of an image that is not a visualization (Garreton et al., 2015).

Most research and design guidelines involving data visualizations predominantly focus on Western, Educated, Industrialized, Rich, and Democratic (WEIRD) cultures. However, both the language surrounding the visualizations and stylistic decisions such as color choice can

influence how the visualization is interpreted (Rakotondravony et al., 2022). With specific words (Chabal & Marian, 2015) and colors (Laurence et al, 1991) having connotations that may be specific to a single language or culture, focusing exclusively on WEIRD countries hinders insight into studying and improving data and visualization practices in non-WEIRD cultures (Rakotondravony et al., 2023).

Objectives

The goal of this project was to investigate visualization design and color significance in different visualization types across different topics from non-western countries. To achieve this goal, the team aimed to collect visualizations, analyze the design choices that went into them, and display our results for group analysis. We gathered visualizations from different source types – social media, news, and journalism – from three countries: Ecuador, Nigeria, and Indonesia. We decided on color as a design element to focus on and developed a program to extract this information. Finally, we created a dashboard website to display and navigate the collected visualizations, their metadata, and color summary.

Collecting and Storing Images

A prerequisite to our first objective was to decide which country to focus on. Based on team interest, familiarity, and fluency in the primary language, we decided on Ecuador. We used search engines to locate Ecuadorian journal articles and social media posts containing some form of visualizations. Alongside images of the actual visualizations, we documented additional information about the visualization and its source, including the platform, source type

and name, year published, visualization type, data source, and topic. Over two weeks, we were able to collect 50 visualizations.

After the initial round of collection, we investigated more efficient ways of collecting and storing the images. On the storage side, we considered several options for a database to use, ultimately settling on MongoDB since it was free and easy to navigate. For collection, we looked into Web scraping, but after encountering major issues (including Instagram's terms of service that forbid scraping) we ended up scrapping the idea.

Later on in the project, we decided to collect visualizations from more countries besides Ecuador. We chose Indonesia and Nigeria; a key factor in our decision was that the three countries from which we collected visualizations were on separate continents. We used the same process for collecting data visualizations from Indonesia and Nigeria as with Ecuador, ending up with 147 visualizations across all three countries.

Analyzing the Collected Visualizations

We created a tool to extract color data from the visualizations collected, counting up the pixels of each color in the image, which we call the palletizer. The palletizer is written in Python, with a QT graphical user interface (GUI). QT was chosen over TKinter (Python's de facto GUI) because QT had more features, was easier to work with, and functioned across multiple operating systems.

Using the GUI, the user can select one or more data visualizations stored as PNG or JPEG images, either from MongoDB or from the computer on which the palletizer is being run. The GUI contains a display of one of the selected visualizations, chosen by the user, a pie chart denoting the most common colors in the image, and a slider allowing the user to choose how

many colors to display, in addition to the basic functionality of opening a visualization and exporting its color data. When the color data is exported from the palletizer, it is generated as two CSV files: one with the total pixel count of the image, and one with the pixel count for each color hex value; these files can then be uploaded back to the database.

Displaying the Results

To display the results of our analysis and to ease additional analysis and pattern searching, we created an online dashboard that pulls the visualizations from the database and displays them side by side in a grid of thumbnails. Users can click on a thumbnail to view a larger version of the same visualization, a pie chart representation of the palletizer output, and the data about the visualization and its source that we collected alongside the visualization. The website also allows filtering by most categories of collected data – filter options are dynamically generated based on what filterable data is attached to the visualizations in the database. Multiple filters can be applied at once; they are combined inclusively if they refer to the same data type, or exclusively otherwise.

Findings and Conclusion

Our original plan for this project was to analyze the visualization practices of many countries, but given the scope and timeline of the project, we adjusted the scope and focused on analyzing the visualizations collected from Nigeria, Ecuador, and Indonesia. Our test dataset contained visualizations relating to a variety of topics including but not limited to COVID-19 and crime. In addition, the types of visualizations varied from bar and line graphs to information graphics and maps. Across the test dataset, we observed that blues, grays, and reds were

predominant across all three countries and all visualization types. Also, bar charts were prevalent in Nigeria and Ecuador, making up 56% and 59% of the sample respectively, while Indonesia had more line graphs, making up 46% of the visualizations collected from the country.

Our team investigated the use of color in data visualizations across non-western countries. By shifting the focus beyond WEIRD cultures, we noted insights offering a distinct perspective to the field of data visualization. Through data collection, processing, and presentation, we provide a platform to showcase different approaches to visualization design across different countries and topics. Our findings highlight the importance of embracing cultural diversity and inclusivity in the development of data visualization techniques and tools. It is vital for ensuring that data visualization practices reflect the complexity of global perspectives.

Recommendations

While working on the project, we found several areas which restricted the scalability of our tools. The free version of MongoDB, which we are using to store our collected visualizations, has a storage limit of 5GB. We didn't hit the limit, but in order to analyze data visualizations on a larger scale, future researchers may need to secure funding to use for unlimited storage or find another platform for storage purposes. The dashboard currently needs to pull all visualizations from MongoDB and store them on the back end in order to function, and only updates its listing when rebooted; a more efficient implementation could use MongoDB's API to download visualizations in real time on an as-needed basis. Filtering could also be accomplished through MongoDB instead of on the dashboard's back end.

A future team could also work to develop features or tools that we couldn't make work, or scrapped due to time limitations. If a web scraping tool could be made to work, it would save

time and effort in the collection process. The palletizer could be improved on the data side by more fully supporting multiple image selection, identifying the text and background colors and separating them from the other colors present, and determining and exporting more data beyond the prevalence of each color within the image. Other useful features that could be added include automatically pulling images from the database, cumulative visualizations of the color data, caching when switching images, and support for other color spaces such as HSL. On the dashboard, cut features include ordering visualizations based on their filterable data, tooltips in the representation of the palletizer output, and support for cumulative visualizations of all entries' data.

Furthermore, the dashboard should be tested for compatibility with assistive technologies such as screen readers.

Authorship Table

Section	Primary Author	Editor(s)
	KB	KB
	VR	KB, JB
	VR	KB, JB
	KB	KB
	KB	KB
	KB	KB, JB
	KB	KB
	JB, KB	KB
	KB	KB
	JB, KB	KB
	KB	KB, JB
	JB	KB
	JB	KB
	KB	KB
	KB	KB, VR
	KB	KB
	VR	KB
	VR	KB
	VR	KB
	VR	KB
	KB	KB, JB
	KB	KB
	KB	KB, JB
	JB	KB
	KB	KB
	JB	KB
	KB	KB
	VR	KB

	JB	KB

Table of Contents

List of Figures

1. Introduction

Data visualizations are an important tool for conveying information in a relatively easy-to-digest format. The variance in color in each image and how they are presented is expansive and that variance is impacted by cultures and regions. In this project explored the differences in the visualizations and analyzed trends in the colors used. To that end, we have created a process and set of tools for the collection, storage, processing, and display of visualizations and their color data trends.

Motivated by improving our understanding of color usage in data visualizations across diverse cultural landscapes, our project set out to provide a pipeline for exploring cultural variance in data visualization. Through this pipeline, we aimed to gather insights into color usage in visualizations and to create a foundation for future projects to branch out from.

We created a process for collecting data visualizations and a script for scraping images off of websites and social media platforms and then storing them in a MongoDB database. The images can then be loaded into a palletizer and processed for their raw color data which can then be pushed back to the database where a website can pull that data and the associated images for analysis and display.

During this four-term project, the team went through multiple iterations of each aspect of the project. The trials and errors encountered by the team throughout each iteration resulted in a final product that is expansive while allowing for growth and customization. In this report, we expand on our objectives and the course of action we took to achieve these. We also present our findings on trends across visualizations from Ecuador, Indonesia, and Nigeria.

2. Background

In this section, we explore the background material that impacted our project. We also discuss the roles of data visualizations in media and journalism and how they are used to impact the delivery of information. We also discuss design, culture, and region and how those factors impact the effectiveness of the information the visualizations are attempting to convey.

2.1 Data Visualizations in the Media

The impact of data visualizations varies based on intent, but the function remains the same. When a visualization is used in a form of media, it is meant to convey some information to the viewer, often visualizations can be used to affect the audience's opinion; be it positively or negatively and are often more impactful than just an image that is not a visualization (Garreton et al., 2015). These visualizations are important because to an uneducated audience, they can make a platform seem more legitimate, and to an educated audience, they can provide a resource for confirming the material being presented.

Data visualizations are used across many types of media from social media platforms such as X (formerly known as Twitter), Facebook, and Instagram to Newspapers – digital and print – and magazines. The visualizations offer a visual format for audiences to see where conclusions are being drawn from and even identify information such as trends and patterns in the data that is not discussed in the context that the visualization is provided. With English being a very prevalent language used for research, design guidelines for data visualizations often focus on Western populations (Sturm et al., 2015). However, different cultures and languages have adopted their own techniques in the manner they display data visualizations. The different ways that visualizations are styled in different locales influence how the visualizations are perceived.

For example, generally, in Western culture, green is associated with good things while red and yellow are associated with danger and caution (Kawai et al., 2022). One study found that viewers "form a reliable opinion of an infographics' appeal within the first 500ms of seeing it," supporting the importance of colors and aesthetics in how visualizations are perceived (Harrison et al., 2015).

Furthermore, it is important to not only consider the visual elements, but also the linguistic context surrounding a data visualization. Language plays a part in helping viewers interpret visualizations (Stofer, 2016). Each language has its own unique characteristics, which can influence viewers' ability to comprehend a visualization. For instance, certain words can carry distinct connotations across languages, which can impact a visualization's message (Chabal & Marian, 2015). Therefore, being knowledgeable of a visualization's target audience's language is essential to ensure information is being communicated accurately.

2.2 Data Visualization Design Effectiveness

The goal of data visualizations is to communicate the importance of different design elements effectively, so the viewer knows where to focus their attention and how to interpret the meaning of the overall design (Bylinskii et al., 2017). Understanding how viewers perceive a design is incredibly useful in ensuring your point comes across. Data visualizations are a powerful tool to communicate insights, trends, and patterns from complex data. Not all data visualizations are equally as effective in showcasing information across different cultures and languages.

Currently, most research into data visualizations has focused on Western, Educated, Industrialized, Rich, and Democratic (WEIRD) cultures shown in Figure 1 in red.
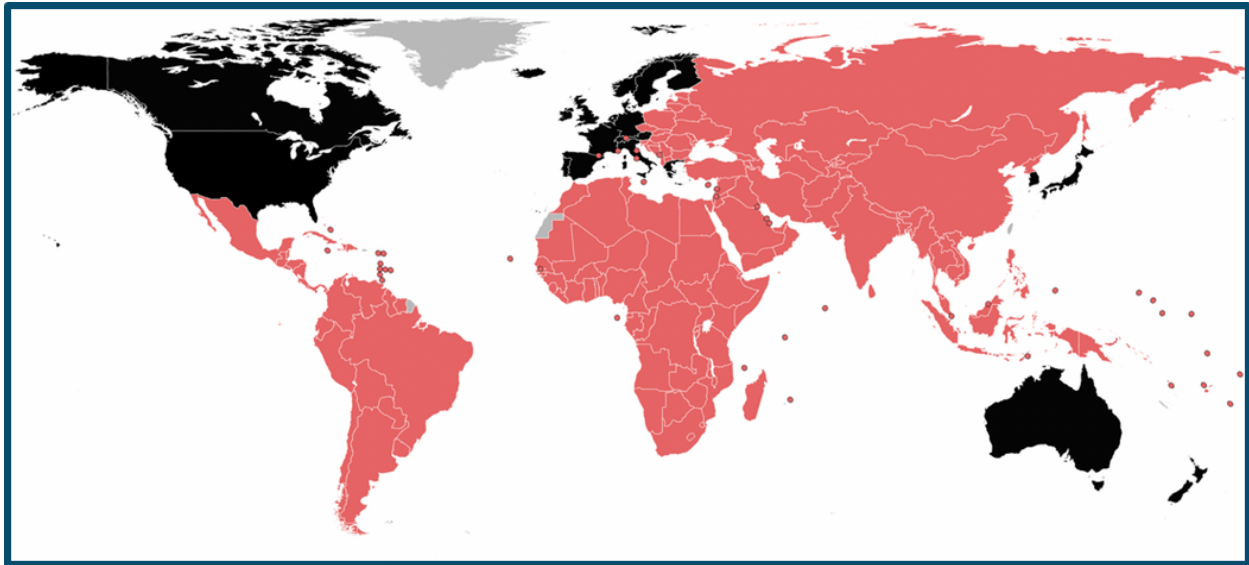
Figure 1: A map of non- Western, Educated, Industrialized, Rich, and Democratic (WEIRD) countries in red.

Having this be the focus has become an issue as it hinders insight into the data practices of the rest of the world that is not a part of WEIRD. When discussing data visualizations, cultural competency is of great importance. Cultural competency is the ability to engage with diverse stakeholders to "include cultural and contextual dimensions important to the evaluation" (American Evaluation Association, 2011). Data visualizations can be interpreted differently due to cultural differences, as a result, it is important to cater a visualization to its audience and take their culture into consideration.

2.3 Aspects of Visualizations to Consider

When considering data visualizations and their effects on viewers there are many aspects to take into account. For example, the language of the text included with a visualization can imply or lead the reader towards certain emotions or ideas, the source the visualization is found in can influence the reader's perception of the data, and the colors used are a key factor in how

information is received. Due to the scope of this project the team prioritized color analysis as the main aspect of visualizations to consider.

The colors used in data visualizations are not solely for aesthetics, they are a crucial tool to convey quantitative information (Simmon, n.d.). Color can be used to highlight relationships, and when properly selected, it can create graphics that tell a story. However, a single color can have different meanings across different cultures. Therefore, it is important to know the audience of your visualization since a clash of colors or meanings can conflict with the objectives sought by the visualization.

In a study done on cross-culture color comparisons, researchers asked students from Japan, China, South Korea, and the United States to state which one of eight colors they associated with 13 words commonly used to describe consumer products (Laurence et al, 1991). The results from this study indicate some similarities and differences across cultures. While subjects from Japan, China, and South Korea associated purple with the word "expensive", subjects from the United States associated purple with "inexpensive". Whereas all four cultures associated blue with high quality and black with power. Research has been done to explore color preferences amongst people, yet there remains more to explore when it comes to the nuanced meanings attached to colors.

Color combinations are considered culturally bound with certain ideologies and traditions (Geboy, 1996). The combination of certain colors can convey distinct meanings based on the color pairing (Madden et al, 2000). For instance, white and red signify life in Japan, while in Mexico it signifies the sacred church. Color usage is crucial in data visualization and must be used wisely. Since colors carry cultural and contextual meanings that can influence how information is perceived, it is important to cater to the visualization's audience. Understanding a

culture's color associations helps in creating data visualizations that resonate effectively with a specific audience.

3. Objectives

The goal of this project was to investigate visualization design and color significance in different visualization types across different topics from non-western countries. To achieve this goal, the team developed three objectives:

1. Collect social media posts and news/journalism sources of data visualizations,
2. Process collected data visualizations,
3. Develop a dashboard or website to present findings.

This chapter describes the approach the team took towards collecting, processing, and displaying data visualizations from non-western countries.

In order to grasp a thorough understanding of how aspects of data visualizations in non-western countries differ from the western design standards, the team chose to gather visualizations from different source types. This included different social media platforms as well as news and journalism sources. We collected data visualizations from three different countries across three different continents: Ecuador, Nigeria, and Indonesia.

The team considered different aspects of data visualizations to analyze and compare. Upon brainstorming, we settled on color to be the feature on which we focused our process and analysis. Our aim was to develop a program to extract the colors used in each data visualization in order to be able to draw comparisons.

To present our findings, the team decided to create a website or dashboard. This user interface would display the collected data visualizations, along with information about them and a color summary. Our goal was for this to be a tool for others to use to collect non-western visualizations and to draw information and conclusions from that collection.

4. Collecting and Storing Images

Because our project focuses on analyzing qualities of data visualizations from non-western countries, the team had to select the country we would first focus on. The criteria we used to select a country consisted of team interest, and team fluency in the primary language. Through brainstorming and discussion, the team began the project by collecting data visualizations from Ecuador. The method by which the team collected data visualizations consisted of using search engines to locate articles and social media posts containing these. Search terms included *"Ecuador data visualization," "Visualización de datos Ecuador,"* and *"Ecuador gráfico."* Below is a collage of the visualization images collected in our initial trial run for Ecuador. As can be seen, a variety of images and visualization types were collected.

Figure 2: A snippet of the spreadsheet used to document data visualizations from Ecuador.

To document the data visualizations, the team used a Google spreadsheet – see Figure 3. As we found data visualizations, the team would characterize each visualization with the following variables: name: name of the visualization, country: the location that the visualization was sourced from, platform: the source that the visualization originated from, source type: the type of media platform that the visualization originated from, source name: the name of the social media, year published: the year the visualization was published on the platform it was collected from, data source: where the data used for the visualization originated from, visualization type: the type of visualization e.g. bar graph, information graphic etc.; topic: the topic of the visualization e.g. COVID-19, crime, etc.; tags: associated tags for the visualization; search terms: the terms used to search for the visualization; description: what the visualization is about; image: image type; and source URL: the URL of where the visualization was sourced. Our initial goal was to collect 50 data visualizations and did so within two weeks.

	A	B	C	D	E	F	G
1	**Media**	**Source Name**	**Type of Source**	**Data Source**	**Link**	**Type of Vis**	**Topic**
27	Website	El Universo	Journalism	Government Ministry	https://www.eluniverso	Bar graph	Crime
28	X	@MashiRafael	Government	Banco Mundial	https://x.com/MashiRa	Bar graph, Line g	Crime
29	Website	Primicias	Journalism	Ministry of Interior	https://www.primicias.	Pie chart, Bar gra	Homicides
30	Facebook	Fiscalia General del Estado Ecuador	Government	Comision Especial de Es	https://www.facebook.	Pie chart, Bar gra	Crime
31	X	Fiscalia Ecuador	Government	Comision Especial de Es	https://x.com/Fiscalia	Pie chart, Timelir	Femicide
32	Website	El Universo	Journalism	Policia Nacional	https://www.eluniverso	Bar graphs	Crime

Figure 3: A snippet of the spreadsheet used to document data visualizations from Ecuador.

After the first round of collecting data visualizations, the team needed to find a more efficient way to store the images. At the time, we were storing images of the visualizations in a Google Drive folder and had an accompanying spreadsheet containing the appropriate details. We decided to begin exploring different database software to store our visualizations and their

data. Simultaneously, we began looking for methods to collect data visualizations more efficiently.

To automate the visualization collection process, we wrote a Python program that would scrape websites with the intention of collecting the data visualizations on the site. However, difficulties arose as the team delved into this matter. X and Instagram both have strict policies against scraping, and therefore we couldn't scrape visualizations from these sources. We also faced difficulties since the website scraper tool we developed proved to not be one-size fits all, meaning it would work with some websites, but not with others. After several unsuccessful trials of using our website scraper, the team decided to move on from this idea.

We then began a process for selecting a database to use to store our data. Our main criteria consisted of price and ability to populate our project website. We trial-ran an Amazon Web Services (AWS) database, however it proved to be too costly. We then considered using OneDrive due to the immense amount of storage it would provide if we used it through our university accounts. However, since we were attempting to connect to a OneDrive folder linked to our WPI school accounts, the school's network security did not allow the team to do this due to authorization requirements. We then attempted to use MongoDB, which was both free and easy to navigate. Given this, the team decided to upload our data visualizations and their accompanying data into a MongoDB cluster. We implemented an embedded JavaScript (ejs) program to help us easily upload to our MongoDB database and view the contents of it in real time. This program would later help us in creating our website deliverable.

The ejs program to upload to the database was a form that asked for image title, image description, country, medium/platform, source type, source name, year published, data source, visualization type, topic, tags, search terms, source URL, image document, and the csv file with

the color breakdown. After successfully completing the form and clicking submit, the team could

scroll down to the Uploaded Images section and see as the data visualization was uploaded

successfully to the database. Figure 4 shows the UI of the online form the team created.

Figure 4: UI of the form used to upload data visualizations to Mongodb.

After having successfully collected data visualizations from Ecuador and uploaded these

to our database, the team decided to conduct two additional data visualization collection rounds.

For our two later trials, we decided to collect data visualization from Indonesia and Nigeria as these two countries were in different continents we had not represented in our project yet and were of interest for the team. We used the same process for collecting data visualizations from Indonesia and Nigeria, as we did with Ecuador. At the end of the three data collection rounds, the team had collected a total of 147 data visualizations. To keep track of the data visualizations from round 2 and 3, we created separate Google spreadsheets for Indonesia and Nigeria to document the data visualizations from each. After this was complete, we then uploaded the visualizations to our database to later analyze.

4.1 Visualization Collection and Storing Challenges

The method of collecting data visualizations posed its own set of challenges. Despite the team using search engines and key phrases to locate data visualizations for our project, it proved to be a meticulous process. From having to document the relevant information pertaining to each visualization, to ensuring that we had not documented the same data visualization twice, this was an arduous process. As the volume of collected data visualizations increased, there was a greater need for a more efficient data collection and storing process.

However, the team encountered further obstacles with the Python program built to scrape websites for data visualizations. During the first trial, the program downloaded all images from the website including logos and ad images. To only get the images from the article, we modified the program to download images of a certain size. However, this proved to be a more arduous task than expected since we would have to sort through the images to find the data visualization we were looking to analyze. The program was also only successful in scraping the data visualization images from certain websites; some websites had the data visualizations as frames,

while others did not allow for the website to be scraped. Additionally, our visualization sources were not just websites with news articles or research papers; we also attempted to collect visualizations from social media. We encountered limitations, such as Instagram's terms of service did not allow for image scraping otherwise your account would be banned.

Additionally, finding a suitable database for our project presented challenges. We needed to ensure it was not costly, preferably with a free option to test it, as well as being able to integrate with our dashboard.

5. Collecting Color Data from Collected Visualizations

5.1 Extracting Color Data

To collect the data needed to analyze the data visualizations collected, we created a palletizer tool. The purpose of this tool was to extract useful data from the images collected focusing on colors represented in the image and pixel counts of each color. This program uses a QT graphical user interface (GUI) that allows for the user to interact with the visualization and determine certain aspects of the outputs. When the Python script is run, the following pop-up displays the GUI created, see Figure 5.
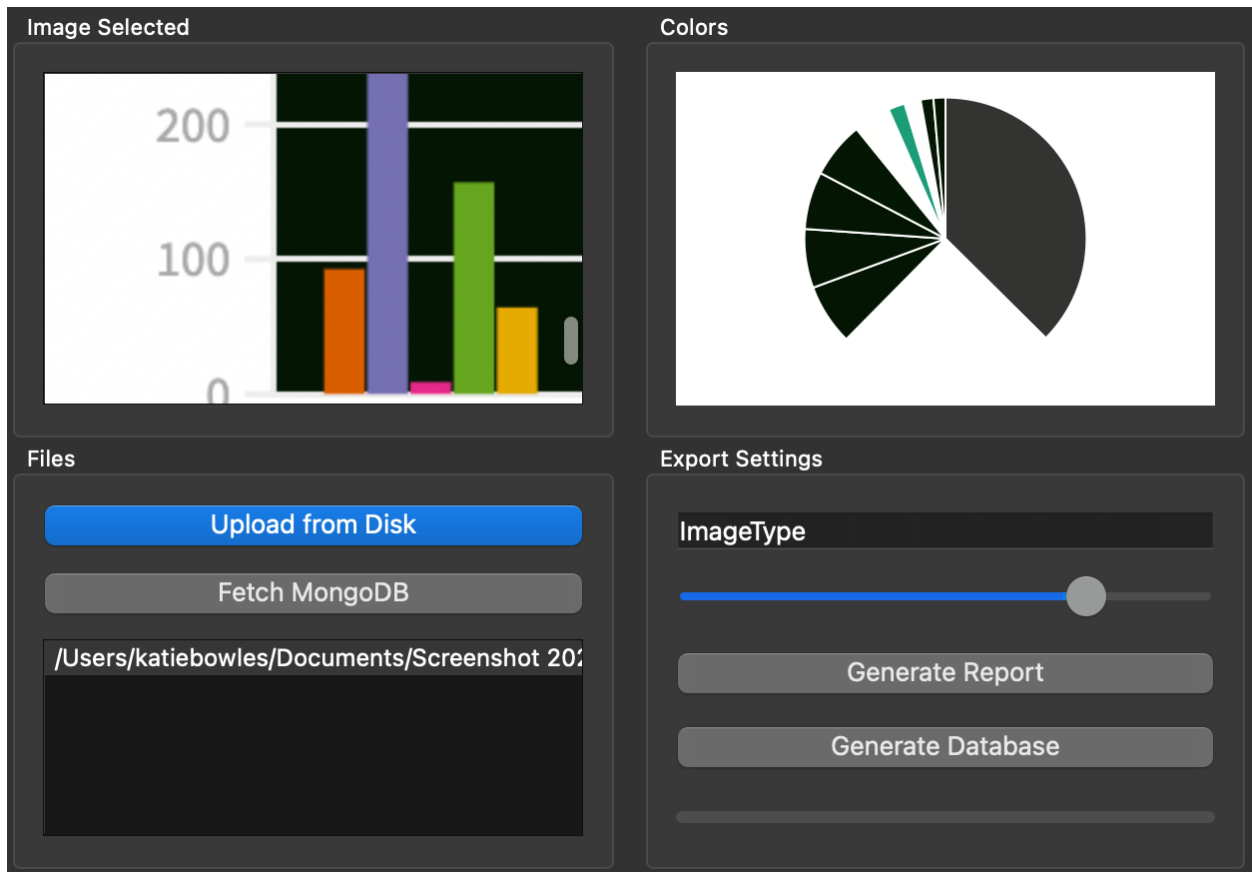
Figure 5: Palletizer user interface showing an example of a data visualization being processed. This palletizer was used to collect raw color data by collecting the hex value of each pixel within the image.

In this window, the user can select one or multiple data visualizations of file png, jpg, or jpeg from their location of choice. These files are listed after selection in the Files box on the bottom left of the pop-up window. Selecting one of the files listed in the Files box will display the image in the Image Selected box located on the top left. When an image is displayed on the left a pie chart in the Colors box on the top right displays n-colors found in the image based on pixel count found in the image in their respective proportions. On the bottom right of the display, is a slider that the user can drag to select the desired quantity of color ranges to be shown in the pie chart. This function and the pie chart are for the back-end user to verify that the program is functioning properly while sorting the colors into their appropriate ranges and are not included in the final CSVs downloaded.

Additionally, the Generate Report button, located on the bottom right of the pop-up, takes the user through a few prompts. This button generates two CSV files. One – shown in Figure 6 – contains the file name, the hex values of the colors in the image, and the count of pixels of occurrences of each color's hex value.

	A	B	C
1	FileName	Color	Occurrences
2	Diversity-info	#20556f	322683
3	Diversity-info	#c5d5de	531166
4	Diversity-info	#123140	80703
5	Diversity-info	#a3b9c4	7901
6	Diversity-info	#7c9caa	154
7	Diversity-info	#5b8195	89
8	Diversity-info	#34667c	272
9	Diversity-info	#2e6179	189
10	Diversity-info	#507a8e	121

Figure 6: Excel format of the first CSV file generated by the palletizer.

The second CSV, the meta report, contains the file name, total pixel count, and file or image type. The file type can be chosen by the back-end user by, before clicking "Generate Report," filling in the "ImageType" box located in the bottom right portion of the pop-up.

Figure 7: Excel format of second CSV file generated by the palletizer.

Both CSVs contain "FileName" to be used as a primary key for creating the front-end dashboard and potential associated visualizations that the user could create.

5.2 Challenges Extracting Color Data

While creating this palletizer tool a few challenges were encountered with two emerging quickly as the main issues. The first and main issue was the color quantization. The range of color values on the RGB and hex scales is quite large and to map them all was beyond the scope of this project's time constraints which led to the questions of should similar colors of a similar value be grouped together and if so, how? The second major challenge encountered was determining which graphical user interface – or GUI – library would be best suited for our end goal and function on different operating systems- OS/Windows/Linux.

To address the issue of color quantization we had to address the question of if colors of a similar hex value should be separated and if so, how? To do this the palletizer was put through a series of tests where an arbitrary color distance function and an arbitrary threshold for the color

grouping were identified based on trial and error. For this testing, and to ensure that future users can test the practicality of the groupings, the pie chart as described in the previous section was created. In addition, given the large amount of color values on the color spectrum, it was found to be far too many color values to display them all in a visualization. When there were too many small counts of values of colors in an image, trying to display those resulted in a section of white in the data visualizations in the palletizer. To combat this, colors whose hex value count of occurrences was less than 20 were excluded from the display; these numbers were included in any total pixel counts while they were excluded from the visualizations. If the user wants to change it, they can easily do so by opening the palletizer python file and changing the number on line 191 pictured here.

```
190        for key, value in self.colorsRaw.items():
191            if value > 20:
```

Figure 8: The lines of Python Code that can be altered to change the colors shown.

Implementing the palletizer with TKinter, a Python GUI toolkit, was the original plan for this project. However, after trial and error, it was found that TKinter, while being Python's de facto GUI, did not function as needed across different versions of OS systems. TKinter is a simpler tool while QT has more available features. This caused a switch to QT. QT allowed for an easier development process and compatibility with more versions of OS operating systems. In addition, the QT interface was cleaner and more scalable than the TKinter interface.

6. Building the Website

To compare different visualizations and search for patterns, we needed a convenient way of showing the visualizations side by side. To that end, we created a simple website to show thumbnails of the visualizations in a grid. As of the writing of this paper, the site is live at https://mqp-visxcultures0.glitch.me/; the code for the website can be found on this project's OSF page at https://osf.io/drzkg/?view_only=67fe375917bf445e80968da0fa5a8b41. In addition to the thumbnails, the website can show the visualizations on a larger scale, a pie chart representing the palletizer output, and various data such as country of origin that we collected alongside the visualizations; the latter can also be used to filter through the visualizations. Finally, near the top of the page is a link leading to a variation of the EJS script used to upload the visualizations to MongoDB.

6.1 Design and Functionality

When the dashboard is opened in a browser, it will by default show all visualizations in the database, limited only by the number displayed at once. They are displayed in a grid in the "Search Results" section at the bottom of the page, and users can click on a thumbnail to open an off canvas containing a larger version of the same visualization, a pie chart representation of the palletizer output, and text denoting the visualization's source URL, country, topic, medium, source type (e.g. journalism, government publishing), source name, data source, year published, visualization type, search terms, and tags. If present, the source URL, and data source function as hyperlinks.
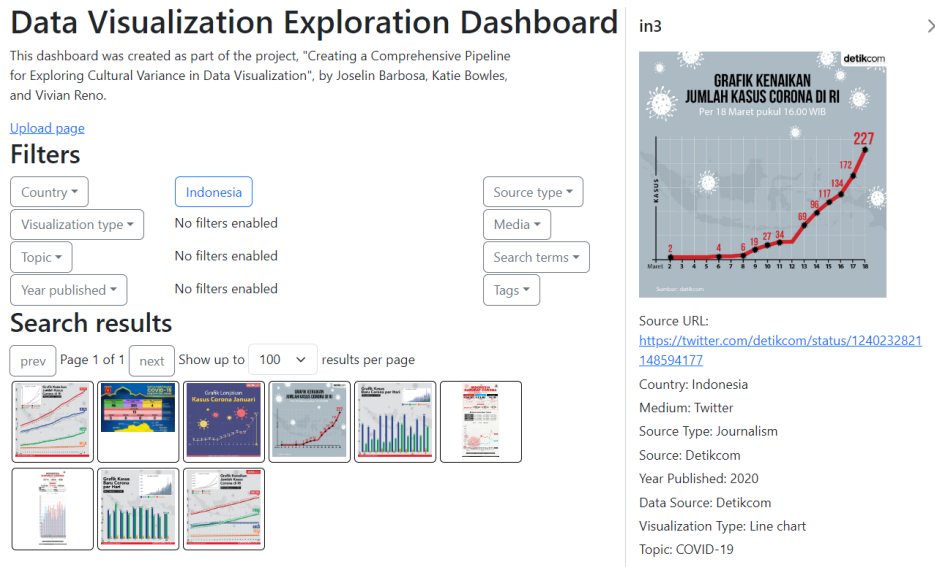
Figure 9: Screenshot of the dashboard with the off canvas open.

To filter the visualizations, users need to open one of the drop-down lists in the "Filters"

section, revealing buttons for every available filter in that category. These buttons start out as

outlines, but when a filter is enabled, its button becomes filled in; additionally, active filters will

also appear outside their corresponding drop down so as to reveal what all filters are enabled at

one time. The filters outside the dropdowns also function as buttons, allowing the user to disable

active filters without opening the dropdowns. Filtering by country, topic, medium, source type,

year published, visualization type, search terms, and tags are supported; filtering by source name

and data source were also implemented but are disabled by default to avoid potentially cluttering

the filter list with many unique source names and data sources.
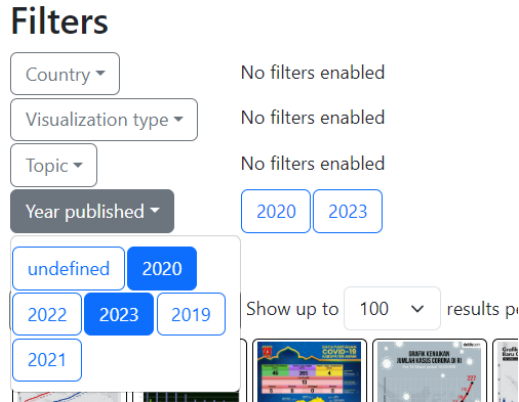
Figure 10: Filter selection UI ("2020" and "2023" are enabled).

When two or more filters within the same category (e.g. published in 2018 and published in 2020) are turned on, the dashboard will show all visualizations which match either filter (in this case, visualizations published in either 2018 or 2020). But when filters in different categories (e.g. bar charts and charts about COVID-19) are selected, only visualizations matching filters in every category with a filter enabled (in this case, bar charts about COVID-19) will be shown.

Figure 11: Results of applying two filters in the same category.

Data Visualization Exploration Dashboard

This dashboard was created as part of the project, "Creating a Comprehensive Pipeline for Exploring Cultural Variance in Data Visualization", by Joselin Barbosa, Katie Bowles, and Vivian Reno.

Upload page

Filters

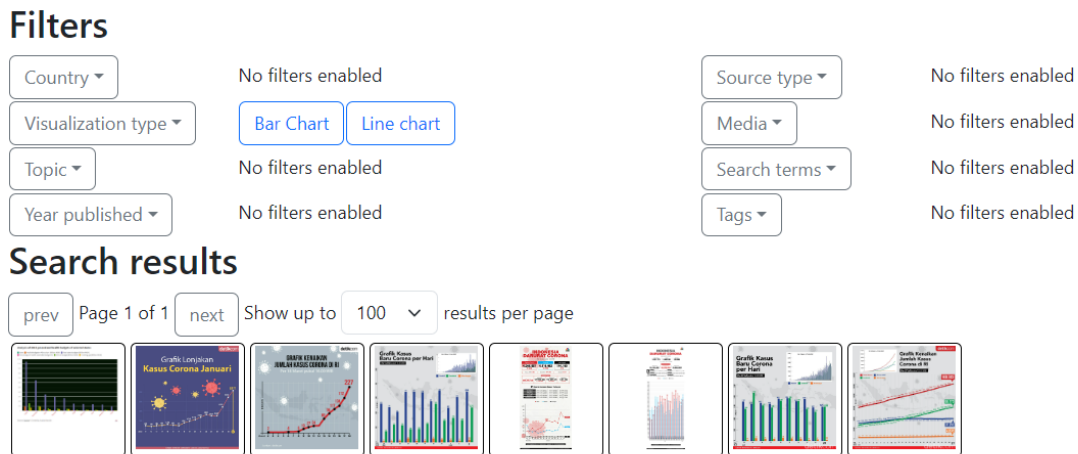|			
Country ▾	No filters enabled	Source type ▾	No filters enabled
Visualization type ▾	No filters enabled	Media ▾	Twitter
Topic ▾	COVID-19	Search terms ▾	No filters enabled
Year published ▾	No filters enabled	Tags ▾	No filters enabled

Search results

prev | Page 1 of 1 | next | Show up to | 20 ▾ | results per page

Figure 12: Results of applying two filters in different categories.

To reduce client memory usage, the visualization grid is paginated. The user can select how many visualizations to show at once, from 10 to 5,000. Default is 100. Buttons for turning the page forward and backward, along with this selection, are in the "Search Results" section alongside the grid.

The website features responsive design; that is, it will change its layout based on the screen size on which it was being viewed. For large screens, the filter selection options will be placed in two columns, but for smaller screens, they will instead form a single column which would become collapsible. The header and visualization grids are not affected by the responsive design; they will simply wrap around the screen as normal. A limitation encountered is that on small screens, the side pane will cover most of the page and would need to collapse before its contents could be changed.

Figure 13: The dashboard as seen on a narrow screen.

6.2 Structure and Implementation

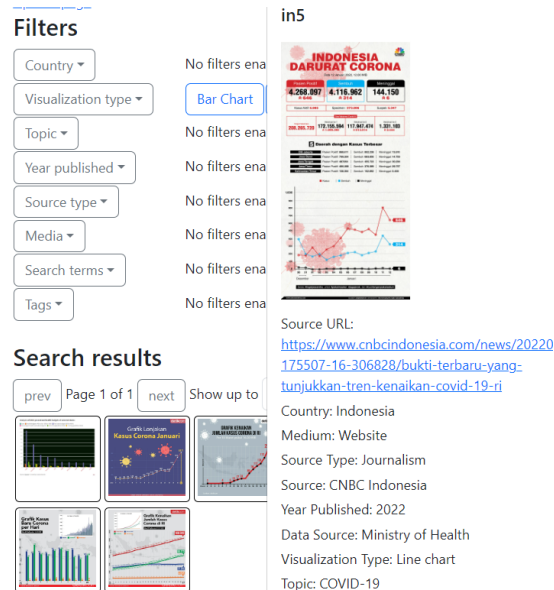Our back end uses Node.js, and the front end uses Bootstrap. The site is hosted using Glitch, separately from the data visualizations themselves, which are stored in MongoDB.

All filtering is performed on the back end, as is all database access. Whenever the active filters are changed, the front end sends a POST request containing a list of the active filters and the number of visualizations, to which the back end returns a list of visualizations which are then displayed on the front end. The back end constantly holds all visualizations' metadata in memory, but the front end keeps only the metadata for the visualizations currently on display. While waiting for the back-end response, the front end temporarily replaces the visualization grid with a Bootstrap spinner, disabling viewing the visualizations' metadata in the meantime, but new filters can still be toggled on or off without waiting.

Filter options are dynamically generated based on the metadata based on the available visualizations. Specifically, after the back end pulls the visualizations and metadata from

MongoDB on startup, it creates a combined list of all metadata filters that are matched by any visualization. Once the front end is also started up, this list is sent to the front end, which dynamically constructs its own filter selection section based on this list. If none of the visualizations have any metadata in a certain category, no filters are available for that category, and the front end says so.

6.3 Website Challenges

Originally, we planned to create the site as a static webpage and host it on GitHub Pages. However, we were unable to find a tutorial for connecting to MongoDB on the front end, so we had to move to a more complex structure with both a back end and front end. This change had the advantage of allowing us to perform filtering on the back end and send over only matching visualizations, saving client resources. But it also had a disadvantage in that GitHub Pages only allows hosting static web pages, not anything requiring a back end. We thus had to investigate other hosting platforms for the website.

Across all team members and our project advisors, we had experience hosting sites using Glitch, AWS, Netlify, and DigitalOcean. To decide which platform to use, we tried to host a very simple website using each platform. As it turned out, Glitch, the first platform we tried using, proved to be so easy to use that we were able to host a fully functional copy of the dashboard with very little effort. The only issue was that Glitch cannot host from a private GitHub repository, meaning we had to copy our code over. The other three platforms were not so friendly to start with. AWS had numerous free and paid plans, with little guidance for beginners with little hosting experience. Netlify specifically required the code be on a platform such as GitHub, requiring a paid plan to host from a private repository. And DigitalOcean's login and

registration site did not even function. With how easy it was to set up the dashboard in Glitch, we chose not to put up with the other platforms' issues.

The UI was initially built using only native JavaScript, HTML, and CSS. A consequence of this implementation was that dropdown menus were hard to create. The solution was to use Bootstrap, as it has built-in dropdown functionality that does not require additional code to be written for such a basic UI component. After the switch to Bootstrap, we revamped the entire UI to use Bootstrap controls over the regular HTML ones, so as to make the appearance of all controls consistent. Another advantage of Bootstrap was the ability to easily create offcanvases (collapsible side panels), one of which was used to display the selected visualization and its metadata. This design solved an issue; the original design called for this information to be displayed on a tooltip which would follow the user's mouse cursor around and update when they roll over a new thumbnail. The old design would have had no touchscreen support and may have been annoying for users. The new design, on the other hand, gives the user better control over the display and fully supports touchscreens, with the only limitation being that the side panel covers the other content on small screens.

7. Conclusion

This section provides recommendations for future improvement of the project based on the challenges outlined in previous chapters, as well as details of our findings achieved in fulfilling our project objectives.

7.1 Findings

Originally this project was intended to contain analysis of many country's visualization practices but given the scope of this project and the available timeline the only analysis that could be done was on the test visualizations collected from Nigeria, Ecuador, and Indonesia. Even though this project was limited in its access to a diverse collection of visualizations, the team was still able to identify some preliminary findings.

7.1.1 Visualization Type and Color Usage

Across the three countries we collected samples from, our findings were that blues, grays, and reds were the most commonly used colors across all countries and visualization types. As can be seen in the figure below containing a set of eight of our sample visualizations, these three-color groups were prevalent among different visualization types.

Figure 14: This image contains a representative portion of the collected visualizations to illustrate the predominant colors found in this project occur in various visualization types.

Figure 15, below, shows the top hex values across all of the sample visualizations collected. This pie chart does exclude the hex value of #ffffff which is the standard white color often used as a background color in images. The pie chart takes the top twenty hex values, minus #ffffff and their occurrences and their proportion compared to each other. As already stated, all of these hex values result in shades of grays and blues – when a larger range of top colors are graphed the next most common colors are shades of red.
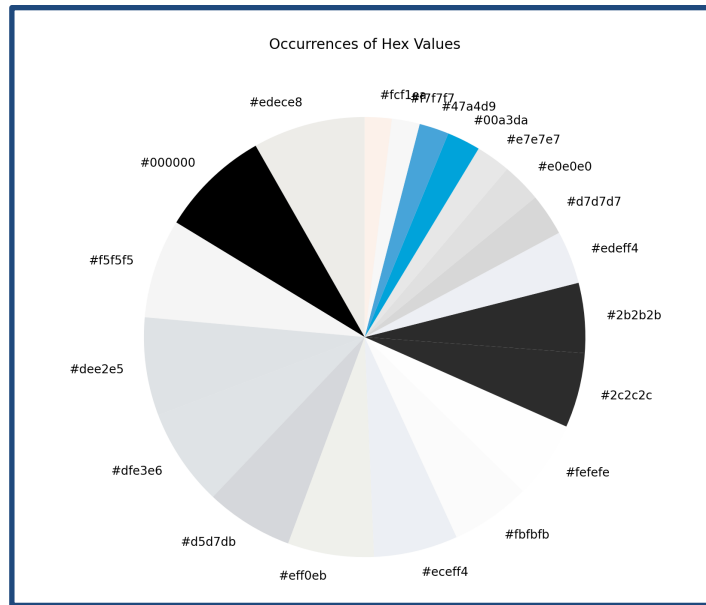
Figure 15: This figure contains the proportions of the top 20 colors across all visualizations collected (excluding hex value #ffffff, white, as it is a common background color for images).

It is important to note that the visualization types varied across the three collections. From Ecuador, the sample size included bar graphs, infographics, line graphs, maps, pie graphs and tables. From Indonesia, the sample size included bar graphs, infographics, line graphs and tables, From Nigeria, the sample size included bar graphs, donut charts, infographics, line charts and pie graphs. Through analyzing the three collections we noted that Ecuador and Nigeria both had predominantly bar graphs: 40% and 54%, respectively. While Indonesia had predominantly line graphs, 50% of the sample visualizations, across all visualization categories. A pie chart of each country's visualization types can be seen in figures 16-18. Even given the limited data set available, these findings show the potential of this project and the tools developed during and how they can be implemented on a larger scale. The distribution of visualization types in each country is shown in the images below.

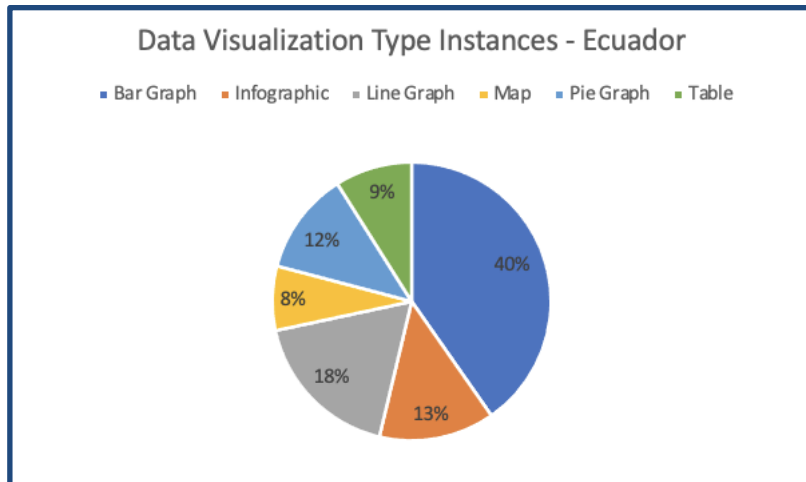*Figure 16: Pie Chart showing the number of different instances of data visualization type from our Ecuadorian sample set.*
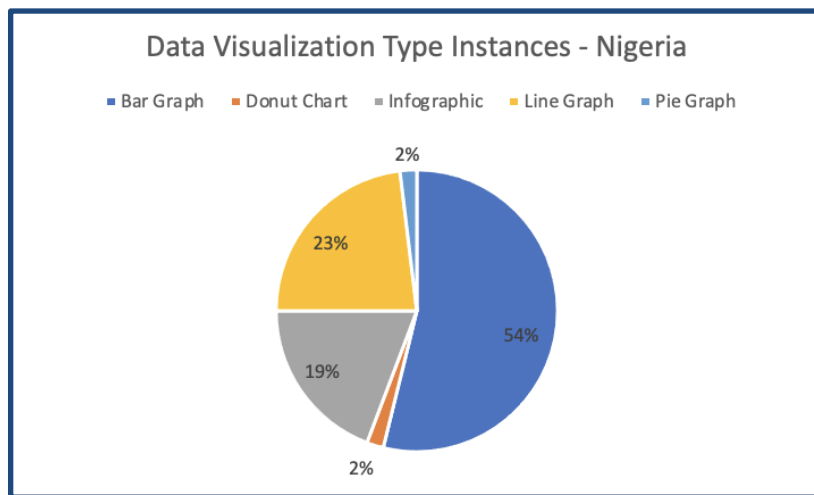

Figure 17: Pie Chart showing the number of different instances of data visualization type from our Nigerian sample set.
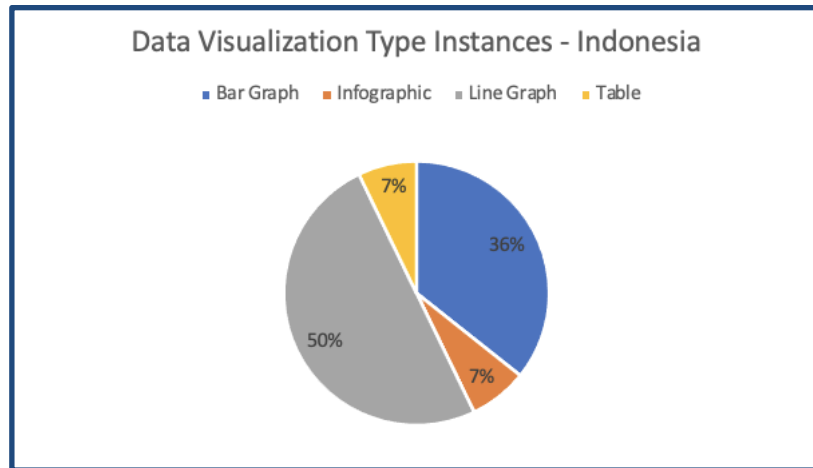
Figure 18: Pie Chart showing the number of different instances of data visualization type from our Indonesian sample set.

7.1.2 Visualization Tools

While not all data visualization sources indicated the tools used to create their visualizations, some did provide such information. From our Ecuadorian collection, news sources tended to maintain their own style of visualization. However, Primicias and Diario Extra, both news outlets in Ecuador, used Flourish and Genially. Flourish is an online platform used to create data visualizations. Whereas Genially is an online tool used for creating interactive content. We also noticed that news outlets were incorporating similar images from what appeared to be a dashboard on COVID and another on femicides. Upon further research, the team discovered that the Ecuadorian government allowed public access to dashboards containing statistics on these topics, which news outlets were then using in their articles.

The Judiciary of Ecuador presented FemicidiosEC, an online tool made publicly available to visualize data on femicide in Ecuador. FemicidiosEC creates dynamic visualizations based on data of femicides and violent killings of women in Ecuador (UNODC, 2022). This tool became a

reality due to a collaboration between the Economic Commission for Latin America and the

Caribbean and the United Nations Office on Drugs and Crime Center of Excellence in Statistical

Information on Government, Crime, Victimization and Justice. This partnership opened the

possibility for future data visualization projects in other regions of Latin America and established

this as a good regional practice.
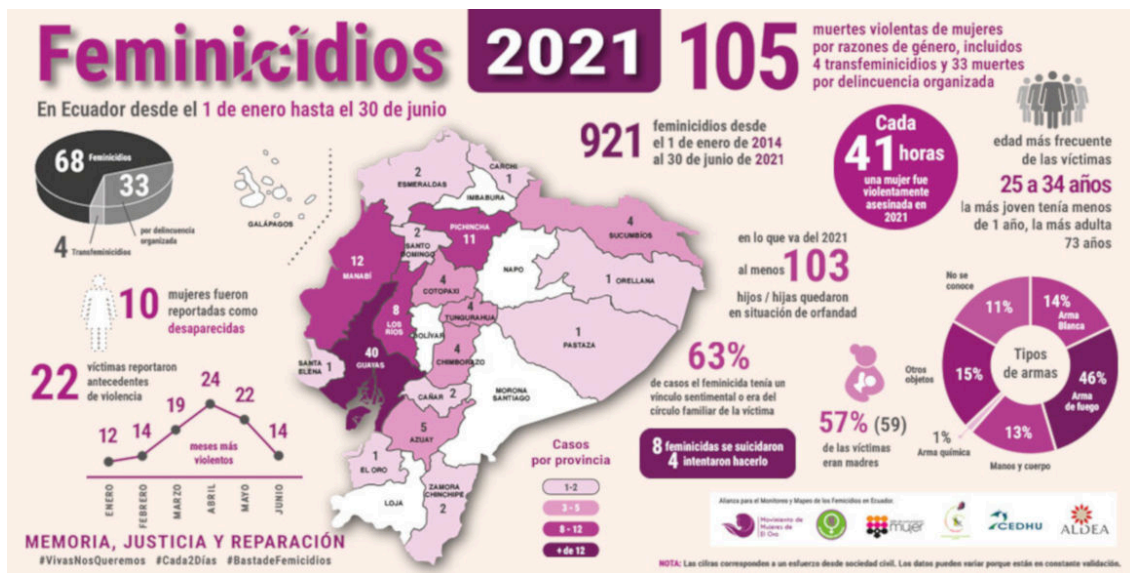
Figure 19: Snippet from Ecuadorian dashboard included by Primicias on femicides in the country.

Like with the Ecuadorian visualizations, there were several Indonesian news outlets that

created their visualizations in house, such as Jakarta Globe, CNBC Indonesia, and Detik News.

Similarly, Indonesia news sources incorporated images from government-made dashboards

containing public information on COVID-19 statistics in the country.

Figure 20: Snippet from Indonesian dashboard included by Jakarta Globe on COVID-19 in the country.

Conversely, in our Nigerian dataset, we noticed more instances of data visualizations made by the news outlets themselves. Additionally, we noted that news outlets, such as This Day Live, used Excel to create their data visualizations (see Figure 21 below).
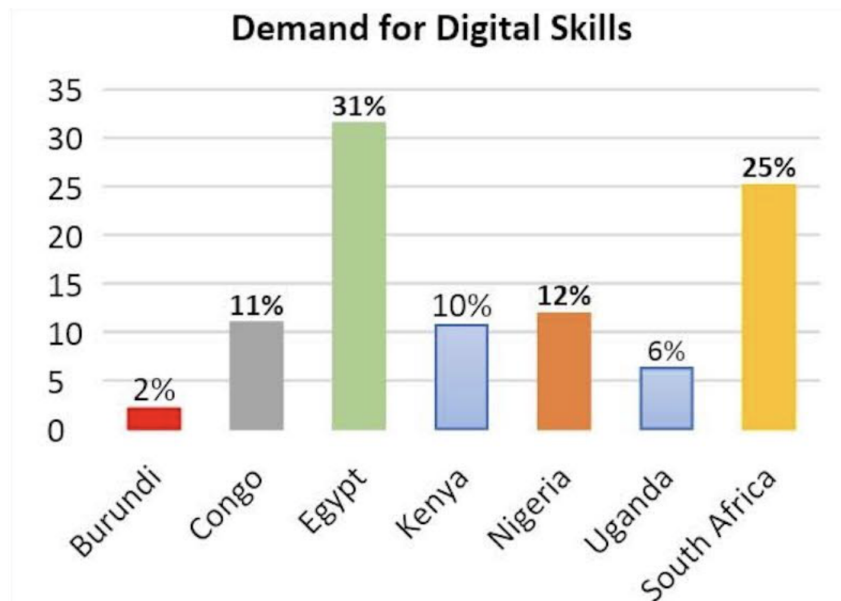

Figure 21: Data visualization featured in This Day Live article on the demand for digital skills in Africa.

7.2 Recommendations

We found throughout the project that there were specific areas of growth that would increase the current function of the tools we created. Below recommendations are provided for the continued growth and development of each of the three main aspects of this project.

7.2.1 Collection Recommendations

Our database currently holds data visualizations for Ecuador, Nigeria, and Indonesia. Future groups can continue to collect data visualizations from non-western countries and add these to the database. We are currently using the free version of Mongodb, and although it has not caused us any troubles yet, we have limited storage of 5GB. It would be worthwhile to gather funding to cover the costs of unlimited storage with Mongodb, or looking for an alternative database that will allow for free unlimited storage.

As for the collection process, more work could be done to improve the workability of the web scraping program. With an effective scraping program that downloads data visualizations from online sources and then uploads these to the database, the team could save time during the collection process.

7.2.2 Palletizer Recommendations

The palletizer was created with the goal of being a scalable tool for the processing of the visualizations collected. There are a plethora of possible areas of improvement that given the scope of this project could not be directly implemented. Many of these were hoped to be completed during the project but given the limited time could not be and others were discovered as the project developed.

There are two ways that the palletizer could be improved. The first direction improvements could take is related to the data. The first of these would be having images be automatically pulled from the internet, reducing a step for the user. Next could be adding support for more than just a MongoDB database so the user can choose a database based on their personal or project preference, adding adjustable distance clustering for the generation of the report so the user can choose the output in the report, which could include dynamic value filtering for the user to choose color values shown, and text filtering support could be added to the palletizer to isolate information related to the words in a visualization. Going forward other possible data related changes are adding more data columns in the CSVs generated so that the user can input more details from the images such as visualization type and making the palletizer more functional with multiple images in general.

The second direction of changes and adjustments is for visualization-related changes. To do so the user could implement a few changes such as implementing other color spaces like RGB and HSL could be added for the user to choose based on their preferences, caching could be added for efficiency when switching between images shown, and lastly, cumulative visualizations and the ability to select multiple images at once could be added as only one visualization can currently be displayed at a time in the palletizer and creates certain limitations for the user..

All these possible changes to the current state of the palletizer are things that would need to be added through manipulation of the base Python script. The team suggests that if changes are added to the palletizer, the ability to function with more than one image displayed in the user interface, and the removal of background color, be prioritized.

7.2.3 Website Recommendations

As with the palletizer, the dashboard also has improvements that were either hoped for and scrapped due to time restrictions or were identified late in the project cycle. Both the back end and front end have possible improvements that could be made.

For the back end, a better solution than what we did would be to use MongoDB's native filtering system, which we found out about late in the project, after filters were already mostly implemented. The current implementation has several flaws that impede scalability which would be addressed by making this change. In our current implementation, the back end pulls every single visualization and its metadata from MongoDB and stores them for the entire time it is running. This is very inefficient in terms of memory usage; while it does not pose a problem with the small size of our sample, it could become an issue when the dashboard is used for a larger sample with thousands of visualizations. The current back end also does not check MongoDB for changes to the database unless it makes a change (by uploading more visualizations) or it is rebooted — an issue if multiple copies of the back end are in use for whatever reason. If we filtered through MongoDB, we should be able to always filter through the most up-to-date version of the data set and do so without loading the entire data set into memory.

The front end could be improved by implementing some features that we were unable to add during the project. Most significantly, we planned on allowing the user to choose one filter category to order search results by (except tags and search terms, since a visualization will typically have multiple of each if it has any). The category would be selected using a <select> menu, like the current number of visualizations per page uses, and would sort the displayed results according to their metadata in that category, displaying rows of visualizations with the same metadata. Another cut feature was tooltips on the pie chart to numerically show the

proportion of the visualization that is each color — the pie chart is generated using d3.js, which does have support for tooltips, but the nature of a pie chart means that the segments are not mapped to data points in the same way as the bars on a bar chart would be, so general d3 tooltip code ends up not working.

Additionally, the front end could be made more accessible. When designing the dashboard, we did not fully consider the use of assistive technologies such as screen readers. The filter selection section is suspect here, since the primary way of checking whether a particular filter is active is to check the button's color (although checking for the second button that appears outside the dropdown for enabled filters is just as accurate, and that button is truly removed when not present.)

7.3 Conclusion

Our team was interested in investigating the use of color in data visualizations across non-western countries. By shifting the focus beyond WEIRD cultures, we were able to note insights that offer a distinct perspective to the field of data visualization. Through data collection, processing, and presentation, we were able to provide a platform to showcase different approaches to visualization design across different countries and topics. Our findings highlight the importance of embracing cultural diversity and inclusivity in the development of data visualization techniques and tools. It is vital for ensuring that data visualization practices reflect the complexity of global perspectives.

Bibliography

Ai He, H., Walny, J., Thoma, S., Willett, W., & Carpendale, S. (2023, July 31). *Enthusiastic and Grounded, Avoidant and Cautious: Understanding Public Receptivity to Data and Visualizations*. OSF. https://osf.io/preprints/osf/mpq32

American Evaluation Association Statement on Cultural Competence in evaluation. American Evaluation Association. (2011). https://www.eval.org/Community/Volunteer/Statement-on-Cultural-Competence-in-Evaluation

Bylinskii, Z., Kim, N. W., O'Donovan, P., Alsheikh, S., Madan, S., Pfister, H., Durand, F., Russell, B., & Hertzmann, A. (2017). Learning Visual Importance for Graphic Designs and Data Visualizations https://doi.org/10.1145/3126594.3126653

Chabal, S., & Marian, V. (2015, June). *Speakers of Different Languages Process the Visual World Differently*. National Library of Medicine. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4451606/

Garreton, M., Morini, F., Celhay, P., Dork, M., & Perra, D. (2015). *Attitudinal effects of data visualizations and illustrations in data stories*. IEEE Transactions on Visualization and Computer Graphics. https://doi.org/10.1109/TVCG.2023.3248319

Harrison, L., Reinecke, K., & Chang, R. (2015, April 1). *Infographic aesthetics: Proceedings of the 33rd annual ACM conference on human factors in computing systems*. ACM Conferences. https://dl.acm.org/doi/abs/10.1145/2702123.2702545

Kawai, C., Zhang, Y., Chu, W., Zheng, C., Gao, C., Gozli, D., Wang, Y., Ansorge, U., &
Lukas, G. (2022, July 15). *The good, the bad, and the red: Implicit color-valence
associations across cultures - psychological research*. SpringerLink.
https://link.springer.com/article/10.1007/s00426-022-01697-5#citeas

Lee, C., Yang, T., Inchoco, G. D., Jones, G. M., & Satyanarayan, A. (2021, May 1). *Viral
visualizations: How coronavirus skeptics use Orthodox data practices to promote
unorthodox science online: Proceedings of the 2021 CHI Conference on Human Factors in
Computing Systems*. ACM Conferences. https://dl.acm.org/doi/10.1145/3411764.3445211

Peck, E. M., Ayuso, S. E., & El-Etr, O. (2019, May 1). *Data is personal: Proceedings of
the 2019 CHI Conference on Human Factors in Computing Systems*. ACM Conferences.
https://dl.acm.org/doi/abs/10.1145/3290605.3300474?casa_token=WZ1pG8fetOwAAAA
A%3AYMI3-aVN5Sdvk-H7PnqOqbsA_9nt1ToRWAyGPOK-o6saPek2O-
6ge8vwt8fNCNVFu8eLDFAg6jpD

Rakotondravony, N., Dhawka, P., & Bancilhon, M. (2023). Beyond English: Centering
Multilingualism in Data Visualization. *Workshop on Data Visualization for Social Good.
arXiv preprint arXiv:2309.06659.*

Rakotondravony, N., Ding, Y., & Harrison, L. (2022). Probablement, Wahrscheinlich,
Likely? a Cross-Language Study of how People Verbalize Probabilities in Icon Array
Visualizations. *IEEE Transactions on Visualization and Computer Graphics,* 29(1), 1189-
1199.

Simmon, R. (n.d.). *Use of color in Data Visualization*. Earth Observatory.

 https://earthobservatory.nasa.gov/resources/blogs/intro_to_color_for_visualization.pdf

Stofer, K. A. (2016). When a Picture Isn't Worth 1000 Words: Learners Struggle to Find

Meaning in Data Visualizations. Journal of Geoscience Education, 64(3), 231–241.

https://doi.org/10.5408/14-053.1

Sturm, C., Oh, A., Linxen, S., Nocera, J. A., Dray, S., & Reinecke, K. (2015, April 1). *How*

weird is HCI?: Proceedings of the 33rd annual ACM conference extended abstracts on

human factors in computing systems. ACM Conferences.

https://dl.acm.org/doi/10.1145/2702613.2702656

UNODC. (2022, January 28). *Ecuador presents femicidiosec, a tool to visualize data of*

femicide and violent killings of women data. - centro de excelencia UNODC. INEGI.

https://www.cdeunodc.inegi.org.mx/index.php/2022/01/28/ecuador-presents-femicidiosec-

a-tool-to-visualize-data-of-femicide-and-violent-killings-of-women-data/