Automatic Interpretation of Map Visualizations with Color-encoded Scalar Values from Bitmap Images

Angela Gabriela Mayhua Quispe

Advisor: Ph.D. Jorge Poco

Committee Members:
Ph.D. Rosane Minghim – Universidade de Sao Paulo – Brazil
Ph.D. Nivan Ferreira – Universidade Federal de Pernambuco – Brazil
Ph.D. Erick Gomez Nieto – Universidad Católica San Pablo – Peru
Ph.D. José Ochoa Luna – Universidad Católica San Pablo – Peru

Thesis submitted to the
Department of Computer Science
in partial fulfillment of the requirements for the degree of
Master in Computer Science.

Universidad Católica San Pablo – UCSP
September of 2018 – Arequipa – Perú

To my parents, Santos and Marina, who never stop giving of themselves in countless ways. To my brothers and sister, who encourage and support me.

Abbreviations

CNN Convolutional Neural Network

CSV Comma-Separated Values

GIS Geographic Information System

HMM Hidden Markov Model

JSON JavaScript Object Notation

ML Machine Learning

MSE Mean Squared Error

MST Minimal Spanning Tree

NLP Natural Language Processing

OCR Optical Character Recognition

RBF Radial Basis Function

SVG Scalable Vector Graphics

SVM Support Vector Machine

Acknowledgments

First and foremost, I want to thank God for having guided me throughout these two years of study.

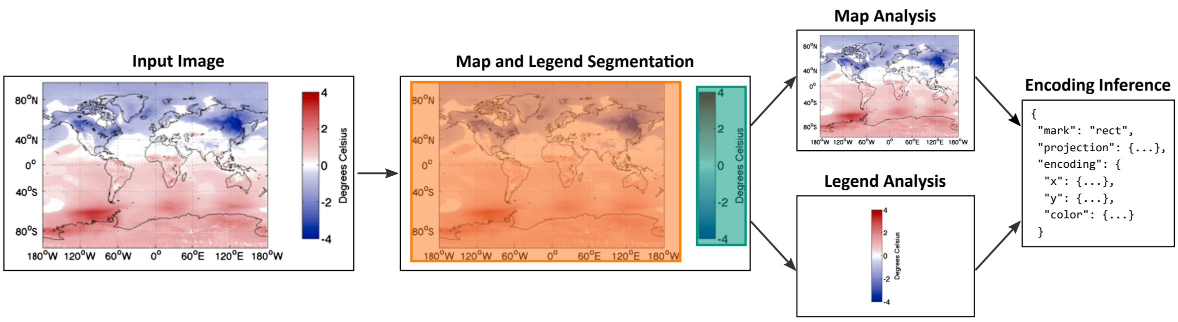
I would like to express my gratitude and appreciation to my advisor Jorge Poco for giving me his guidance and support throughout the master program, for the time dedicated to discuss my doubts and advances, and his patience during the preparation of this thesis.

I also want to thank my parents, sister and brothers, for their constant support and encouragement during these years of study.

I would like to thank in a special way to the National Council for Science, Technology and Technological Innovation (CONCYTEC-PERU) and to the National Fund for Scientific Development, Technological and Technological Innovation (FONDECYT-CIENCIACTIVA), which through the Management Agreement 234-2015-FONDECYT have allowed the grant and financing of my studies in the Master Program in Computer Science at Universidad Católica San Pablo (UCSP).

Finally, I thank all the people who directly or indirectly helped me in the preparation and presentation of this work, either by exchanging ideas, giving me recommendations or encouraging me to continue.

Abstract



Map visualizations are used in diverse domains to show geographic data (*e.g.*, climate research, oceanography, business analyses, *etc.*). These visualizations can be found in news articles, scientific papers, and on the Web. However, many map visualizations are available only as bitmap images, hindering machine interpretation of the visualized data for indexing and reuse.

In this work, we propose a pipeline to recover the visual encodings from bitmap images of geographic maps with color-encoded scalar values. We evaluate our results using map images from scientific documents, achieving high accuracy along each step of the pipeline. In addition, we present iGeoMap, our web-based system that uses the extracted visual encoding to enable user-interaction over bitmap images of map visualizations.

Keywords: Visual encoding, Map interpretation, Map visualization.

Resumen

Las visualizaciones de mapas son usadas en diferentes áreas para mostrar datos geográficos (por ejemplo, datos climatológicos u oceanográficos, resultados de análisis empresariales, entre otros). Estas visualizaciones se pueden encontrar en artículos de noticias, artículos científicos y en la Web; sin embargo, muchas de ellas están disponibles como imágenes en mapa de bits, lo que dificulta que el computador interprete los datos visualizados para su indexación y reutilización.

En este trabajo proponemos una secuencia de pasos para recuperar la codificación visual a partir de imágenes en mapa de bits de mapas geográficos que utilizan el color para codificar los valores de los datos. Nuestros resultados fueron analizados usando mapas extraídos de documentos científicos, logrando una alta precisión en cada paso propuesto. Adicionalmente presentamos a iGeoMap, nuestro sistema web que utiliza la codificación visual extraída para permitir la interacción del usuario sobre imágenes en mapa de bits de visualizaciones de mapas.

Palabras clave: Codificación visual, Interpretación de mapa, Visualización de mapa.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

In Section 1.1 we describe the motivation and context of our work, Section 1.2 presents our problem statement. Section 1.3 shows the objectives of this work and Section 1.4 details our main contributions. Finally, Section 1.5 describes the structure of this thesis document.

1.1 Motivation and Context

Scientific charts are commonly used to visualize quantitative information because they show keypoints and trends among the data. Geographic maps are a popular form of data visualization, used to convey information within a geo-spatial context. The use of maps is not limited to experts such as geographers or cartographers: millions of maps are produced and used by scientists, students, governments, and companies for a variety of analytical purposes (*e.g.*, environmental, economic, political or social). A well-designed map encodes information so as to be interpretable by human viewers; however, these maps are often published as bitmap images, without access to the underlying data. Having access only to pixel-level information impedes automatic processing for tasks such as indexing, search, and analysis (Jung et al., 2017; Siegel et al., 2016) because metadata and pixel values do not include enough information about the content and data plotted on the chart. For that reason, it is difficult to find and reuse map data using either spatial queries (*e.g.*, find all maps involving a specific country) or semantic queries (*e.g.*, find all maps with temperature values in a particular range) (Walter et al., 2013). We need computational solutions to automatically process maps due to the existence of millions of maps that have been digitally scanned or digitally created (Chiang et al., 2014).

Existing methods for automatic chart interpretation focus on analyzing common statistical graphics such as bar, line, area, or pie charts. Some projects attempt to recover the underlying data (Savva et al., 2011; Gao et al., 2012; Al-Zaidy and Giles, 2015; Al-Zaidy et al., 2016; Jung et al., 2017; Siegel et al., 2016; Tummers, 2006), while others focus on recovering the visual encodings (Harper and Agrawala, 2014; Poco and Heer, 2017). However, these systems do not support analysis of geographic maps. In this work, we extend these prior approaches to recover visual encodings for map images with color-encoded scalar values.

Our primary contribution is a map image analysis pipeline that (i) segments an input image into map and legend regions, and then for each region (ii) identifies text elements, extracts their content using Optical Character Recognition (OCR), and classifies their roles (*e.g.*, legend labels, latitude label, longitude label, *etc.*). Next, our system (iii) determines the type of color legend (*e.g.*, continuous or quantized) and (iv) infers the map projection used (*e.g.*, Equirectangular, Miller, or Robinson). We leverage the extracted text, legend information, and map projection to recover a visual encoding specification in a declarative grammar similar to Vega-Lite (Satyanarayan et al., 2017), a high-level grammar of graphics. An additional contribution is a manually-annotated corpus of geographic map images (each containing a color legend) extracted from scientific papers in the field of climate change, which was used to evaluate our pipeline.

We also present a web-based system named iGeoMap that uses the visual encoding inferred by our pipeline to enable user-interaction over bitmap images of map visualizations. The interactions offered by iGeoMap include recoloring, automatic caption generation, map reprojection and data extraction from map visualization images.

1.2 Problem Statement

Nowadays, there is a large number of geographic map images available on scientific papers, news articles and on the Web; however, users do not have access to the underlying data, and to our knowledge, a method to extract the visual encoding from map visualizations does not exist. For that reason, we propose to apply reverse engineering to map visualizations with color-encoded scalar values and generate as output its corresponding visual encoding. In addition, we developed a web-based system that uses the extracted visual encoding to enable user-interaction from bitmap images of map visualizations.

1.3 Objectives

General Objective

Our main objective is to propose a pipeline to infer the visual encoding in JavaScript Object Notation (JSON) format from a map image with color-encoded scalar values.

Specific Objectives

To achieve our main objective, we have the following specific objectives:

- Extract spatial information from the map plotted on the image.
- Extract color information from the color legend on the image.
- Develop applications that use the extracted information and enable user-interaction.
- Evaluate each step of the pipeline using map images from scientific documents.

1.4 Contributions

This thesis proposes a novel map image analysis pipeline to recover the visual encoding from map visualizations with color-encoded scalar values. A map visualization has two important parts that need to be analyzed: geographic map and color legend. Our contributions are related to each part and are detailed below.

- Extracting and Retargeting Color Mappings from Bitmap Images of Visualizations (Poco et al., 2018).
  - We propose a method to semi-automatically extract color encodings from bitmap visualization images. This color mapping is recovered using color and text information from color legend.
  - We also demonstrate the utility of the proposed method through two user-facing applications: automatic recoloring and interactive overlays.
- Extracting Visual Encodings from Map Chart Images with Color-encoded Scalar Values (Mayhua et al., 2018).
  - We propose a map image analysis pipeline that extracts spatial information and color information to recover the visual encoding specification from map images.
  - We also show the usefulness of our proposal through two applications: data extraction and reprojection of map charts.

1.5 Outline

This thesis document is divided into six chapters. After this introduction and problem formulation, in Chapter 2 we survey the literature on map interpretation, automatic chart interpretation and interactive applications from chart images. Chapter 3 presents some basic concepts about the mapping of color and geographic map properties. Next, in Chapter 4 we describe in detail the corpus, techniques used by our pipeline and their evaluation results. Chapter 5 presents our web-based system named iGeoMap and its different modules. Finally, the limitations, future works, and conclusions of this work are presented in Chapter 6.

Chapter 2

Related Work

Our work draws on prior research in the areas of map interpretation that is focused on extracting information from maps, automatic chart interpretation focused on analyzing charts and interactive applications from chart images that enable user-interaction.

2.1 Map Interpretation

Researchers have proposed various methods to perform automatic *map interpretation* (Walter and Luo, 2011) to extract information from maps and analyze their content. For instance, Dhar and Chanda (Dhar and Chanda, 2006) analyze scanned topographic maps to extract and recognize symbols (*e.g.*, trees, forests, rivers, cities, huts, *etc.*) and text contained within the map. One of the steps is to separate the image into four layers: green elements (trees, forests), red elements (streets), blue elements (rivers, lakes) and black elements (text). The map scale and range of latitude/longitude coordinates are entered by the user to locate points on the map given their geographical coordinates. Finally, the output is an *e-map* that can be used as input to Geographic Information System (GIS). Pezeshk and Tutwiler (Pezeshk and Tutwiler, 2011) also worked on scanned topographic maps; their purpose was to automatically extract each component of the map in separate layers and recognize the text contained. They propose an algorithm for extracting linear features to generate a layer containing map lines (streets, roads, *etc.*); they then use the RANSAC algorithm (Fischler and Bolles, 1981) to improve the text preprocessing and a Hidden Markov Model (HMM) to recognize texts and generate a text output layer.

These previous works focus mainly on topographic maps — *i.e.*, maps characterized by contour lines and road lines (Pezeshk and Tutwiler, 2011) — and recognizing their symbols. Our approach automatically extracts spatial information from the geographical map contained in a map visualization; this information includes the type of geographic projection used by the map and the range of latitude and longitude values in the displayed region.

2.2 Automatic Chart Interpretation

A growing number of techniques focus on the "inverse problem" of data visualization: given a visualization, recover the underlying visual encoding and its corresponding data values (Poco et al., 2018). Some of these approaches have focused on *data extraction*. For instance, ReVision (Savva et al., 2011) classifies images by chart type and extracts data from pie and bar charts to output a relational data table. Similarly, the VIEW system (Gao et al., 2012) extracts information from raster-format charts (*e.g.*, pie, bar, and line charts). It first distinguishes graphical and textual connected-components; depending on the graphic type it applies a different approach to extract data; finally, it generates a data table with that information. Al-Zaidy et al. Al-Zaidy et al. (2016) propose a system that extracts data values from bitmap images of bar charts and generates a semantic graph using the label roles (*e.g.*, x-title, x-labels, y-title, *etc.*); then, the semantic graph is used to generate a summary that describes the input image.

FigureSeer (Siegel et al., 2016) is a framework that extracts information from line charts. It detects the axes to extract their labels and infers their scales through curve fitting. To perform legend analysis, it uses a random-forest classifier (Breiman, 2001) to determine whether or not text serves as a legend label and then obtains its symbol. The analysis of the plotting area is done using Support Vector Machine (SVM) (Cortes and Vapnik, 1995) and a Convolutional Neural Network (CNN) (LeCun and Bengio, 1998) to learn functions and avoid problems caused by the occlusion between the lines. Another application is ChartSense (Jung et al., 2017), an interactive system for data extraction from five types of charts: line, area, radar, bar, and pie charts. Its first step is to classify chart images using a classifier based on GoogLeNet (Szegedy et al., 2015); it then extracts the data using optimized extraction algorithms for each chart type. These approaches extract data from charts that contain discrete legends (*e.g.*, bar, pie, area, line, or radar charts). Our work is focused on the extraction of data from visual components in map visualizations that contain continuous and quantized color legends. This chart type has not been addressed so far, despite being considered in ReVision (Savva et al., 2011) during its classification step.

On the other hand, some methods have been focused on *recovering visual encoding* from a chart. Harper and Agrawala (Harper and Agrawala, 2014) present a tool to decompose and redesign visualizations created with the D3 library (Bostock et al., 2011) (*e.g.*, bar charts, line charts, scatter plots, donut charts, and choropleth). This tool extracts data, marks, and visual encoding by analyzing the Scalable Vector Graphics (SVG) elements of the chart and the data bound to those elements via JavaScript. Poco and Heer (Poco and Heer, 2017) propose a method to recover visual encodings from bitmap images of bar charts, area charts, line charts, and scatter plots; their pipeline identifies textual elements in the image, determines their role within the chart (*e.g.*, chart title, x-labels, x-title, *etc.*), and recovers the text content using OCR. They also trained a CNN (LeCun and Bengio, 1998) for classifying 10 chart types, which achieved an average accuracy better than ReVision and ChartSense, achieving an accuracy of 96% for classifying maps. Using this extracted information they then recover a visual encoding specification. However, their work does not include extraction of color encodings or geographic projections.

As part of this thesis, we presented a work (Poco et al., 2018) where we proposed a technique to extract the color encoding from discrete and continuous legends of chart images, including geographic maps. We identify the colors used and the legend texts, then recover the full color mapping (*i.e.*, associating value labels with their corresponding colors). We continue our thesis work upon that approach focusing on map visualizations; thus, we had to tackle other challenges (such as identifying map projections) and develop new applications enabled by our map image analysis pipeline.

2.3 Interactive Applications from Chart Images

The extracted information from chart images can be useful for different applications. For instance, ReVision (Savva et al., 2011) has an interface to redesign the input chart based on the relational data table extracted by its pipeline. Kong and Agrawala (Kong and Agrawala, 2012) propose to create interactive overlays that are placed above chart bitmap images using the extracted data by ReVision (Savva et al., 2011) and Datathief (Tummers, 2006) pipelines to improve the chart reading.

Kong et al. (Kong et al., 2014) developed an interactive document viewer to improve the reading experience, in this application the user can select a paragraph in a document and some components in the charts are highlighted depending on the selection; they use ReVision (Savva et al., 2011) to extract data from bar charts and a manual annotation interface to recover the original data for other chart types. Other works like ChartSense (Jung et al., 2017) and iVoLVER (Méndez et al., 2016) use semi-automatic approaches to extract data values and also present interactive annotation interfaces to correct the output data and improve the interpretation of charts.

In the same way, we propose a web-based system named iGeoMap that enables the user-interaction on bitmap images of map visualizations. iGeoMap uses the visual encoding generated by our pipeline to create interactive overlays, generate automatic captions, recolor and reproject the input map visualization.

2.4 Final Considerations

This chapter presented some recent proposals related to our thesis work. Some research works have been focused on analyzing topographic maps to extract symbols and texts. On the other hand, other works have focused on extracting data from chart images that contain discrete color legends (*e.g.*, bar charts, line charts, pie charts) to improve the chart understanding through interactive applications.

The next chapter will present some concepts needed to understand better our work, those concepts are related to the mapping of color and geographic map properties.

Chapter 3

Background

In this chapter we present basic concepts that are needed to understand better our work. Section 3.1 presents definitions about the mapping of color and Section 3.2 details concepts related to geographic map properties.

3.1 Mapping of Color

A common visual channel used to encode data values in map visualizations is the color; for that reason, in this section, we explain some concepts related to the mapping of color.

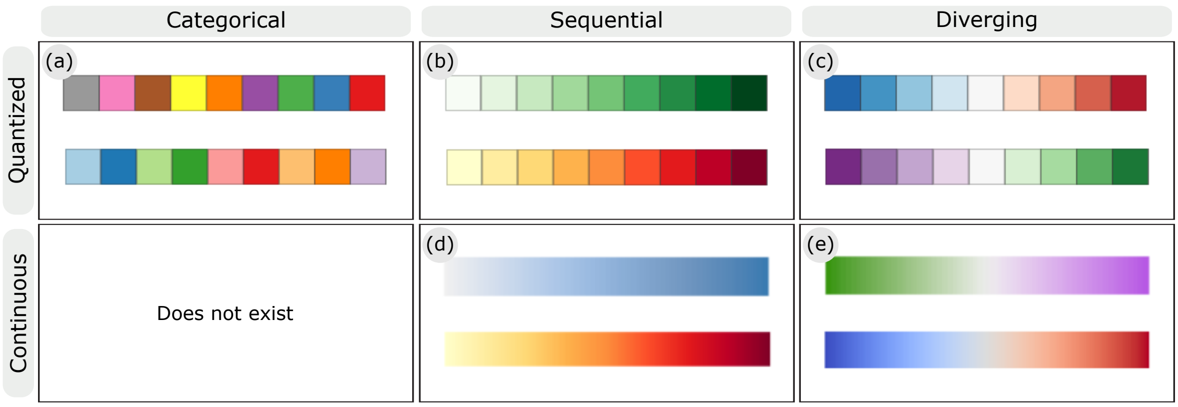


Figure 3.1: Examples of colormap types that show three classes of colormaps: categorical, sequential and diverging. Depending on the data nature, colormaps can be quantized or continuous. (a) - (c) were generated by ColorBrewer (`http://colorbrewer2.org/`), and (d), (e) were retrieved from the user's guide of Matplotlib (`https://matplotlib.org/users/colormaps.html`).

A colormap determines the relationship between colors and data values (Munzner, 2014, Chapter 10). Colormaps can be *categorical*, *sequential* and *diverging* colormaps.

a) **Categorical colormaps:** are suited to representing categorical data, where each color encodes a category or group. There is not an order between the colors (see Figure 3.1a).

b) **Sequential colormaps:** are suited to ordered data that progress from a minimum to a maximum value. As we see in Figure 3.1b and Figure 3.1d, light colors will represent low data values and high data values are represented by dark colors.

c) **Diverging colormaps:** as it is shown in Figure 3.1c and Figure 3.1e, these colormaps have a neutral color (*e.g.*, white, gray) at the middle to indicate the zero point and two dark colors at the endpoints to perceive a negative and positive side.

Depending on the data nature, colormaps can be *continuous* or *quantized* (Munzner, 2014, Chapter 10).

a) **Continuous colormaps:** are used for quantitative data to represent a range of values or a continuous domain (see Figure 3.1d and Figure 3.1e).

b) **Quantized colormaps:** are segmented colormaps into discrete bins of color. This colormap is suitable for categorical data; also can be used for ordinal data to emphasize its discrete nature (see the first row in Figure 3.1).

3.2 Geographic Map Properties

In this section we detail some basic concepts related to geographic maps necessary for the discussion that follows.

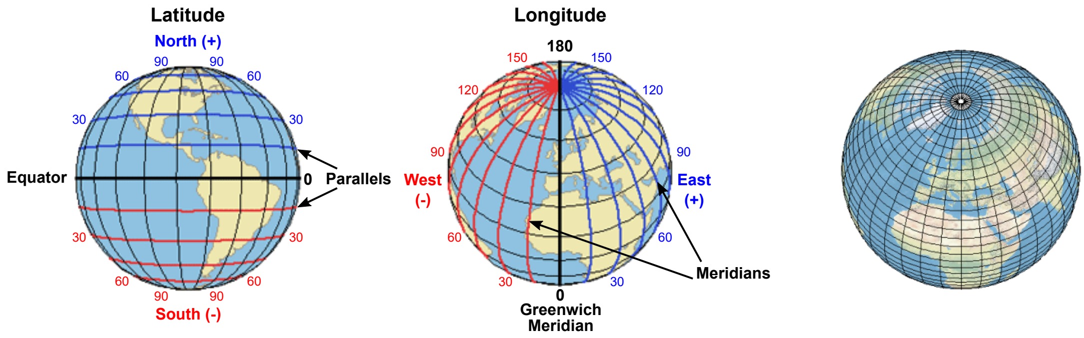
Equator. It is the great circle around the Earth that is equidistant from the geographic poles. The Equator divides the Earth into the Northern and Southern hemispheres.

Parallels. Imaginary lines around the Earth parallel to the Equator. They are numbered by degrees from the Equator to poles (see Figure 3.2a).

Meridians. Imaginary north-south lines on the Earth's surface that connects the geographic poles and are numbered by degrees. The Greenwich Meridian is the zero meridian that passes through Greenwich, a borough of London (see Figure 3.2b).

Latitude. Latitude measures in degrees how far north or south of the Equator a place is located. This value range from $0°$ to (+/-)$90°$. The Equator is situated at $0°$, the North Pole at $+90°$ and the South Pole at $-90°$ (see Figure 3.2a).

Longitude. Longitude measures in degrees how far east or west of the Greenwich meridian a place is located. This value ranges from $0°$ to (+/-)$180°$ (see Figure 3.2b).



(a) Latitude. (b) Longitude. (c) Graticule.

Figure 3.2: Meridians, parallels, latitude, longitude and graticule. (a) Parallels are the imaginary lines parallel to the Equator and latitude is the distance between one parallel to the Equator. (b) Meridians are imaginary lines that connect the poles and longitude is the distance between one meridian to the Greenwich meridian (both figures adapted from `https://iepbachillerato.wordpress.com/latitud-y-longitud/`). (c) The graticule displays both parallels and meridians (from `http://desktop.arcgis.com/en/arcmap/10.3/map/page-layouts/what-are-grids-and-graticules-.htm`).

Graticule. Grid lines that display the parallels and meridians of the Earth. This is used to show a point in geographic coordinates (see Figure 3.2c).

Geographic map projection. As we show in Figure 3.3, a map projection is the transformation of the latitudes and longitudes of locations on the Earth's surface (3D space) into locations on a plane (2D space) (Snyder and Voxland, 1989).

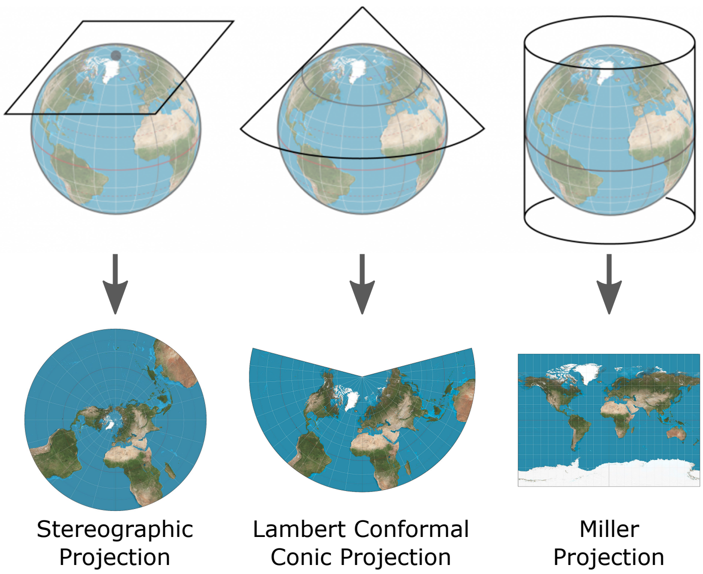


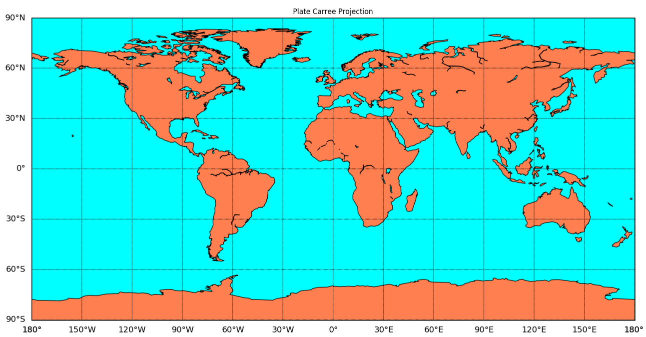
Figure 3.3: A map projection is the transformation of the Earth's surface into a 2D plane (adapted from `https://gistbok.ucgis.org/bok-topics/map-projections`).

Depending on the map projection applied to the Earth's surface, the 2D plane can vary. Below we detail the map projections used in this work.

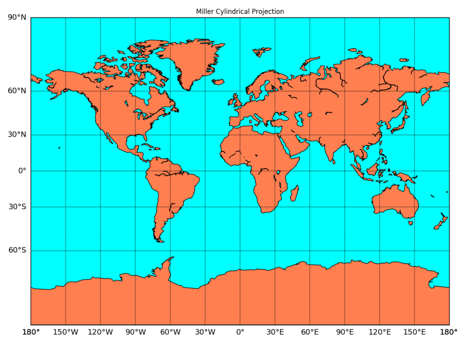
a) **Equirectangular projection:** the meridians and parallels of this projection are equally spaced straight parallel lines (Snyder and Voxland, 1989). The graticule of this projection has rectangles with the same shape, size, and area. This projection is also known as Plate Carre when the central parallel is the Equator, and the rectangles in the graticule are perfect squares. In Figure 3.4a the central parallel is the Equator, and we can see that the distance between 0° and 30°N is equal to the distance from 60°S to 90°S.

b) **Miller projection:** the meridians are straight parallel lines that are equally spaced and their length are equal to 73% of Equator length. The parallels are also straight parallel lines; however, they are unequally spaced, closest near the Equator (Snyder and Voxland, 1989). We see in Figure 3.4b that the distance between parallels is not the same.

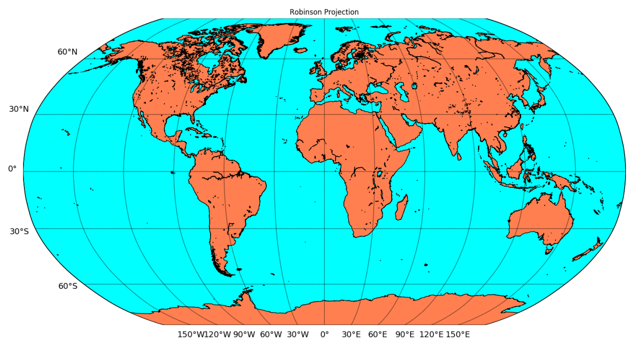
c) **Robinson projection:** the central meridian is a straight line of length equal to 51% of Equator length. Other meridians are elliptical arcs that are equally spaced. The parallels are straight lines that are equally spaced between 38°S and 38°N, and this space decreases beyond these latitudes. Figure 3.4c shows an example of this projection.



(a) Equirectangular projection.



(b) Miller projection.



(c) Robinson projection.

Figure 3.4: Map projections used in this work. (a) Meridians and parallels in the Equirectangular projection are equally spaced straight parallel lines. (b) In Miller projection, the parallels are unequally spaced, closest near the Equator. (c) Robinson projection has elliptical arcs as meridians and the distance between parallels decreases near to the poles. These figures were generated by Basemap Toolkit (Whitaker, 2016)

3.3 Final Considerations

In this chapter, we have presented some concepts about the mapping of color and geographic map properties. Color is used as a visual channel in map visualizations, and the selected colormap type depends mainly on data nature. Furthermore, map projections are used to transform the Earth's 3D surface into a 2D plane.

Next chapter will present our image corpus and how it was created, also presents in detail the steps of our pipeline and their corresponding evaluation results.

Chapter 4

Automatic Interpretation of Map Visualizations

Our work is based on the application of *reverse engineering* to bitmap images of map visualizations to recover their visual encodings in JSON format. To achieve this goal, we extract the spatial and color information from a map visualization image using different techniques, which are presented in this chapter. Section 4.1 describes our corpus and how it was generated. Section 4.2 presents an overview of our work. Sections 4.3, 4.4 and 4.5 detail the techniques used to extract spatial and color information; in addition, each of these sections presents the evaluation results. Finally, Section 4.6 details how the visual encoding is inferred using the information obtained on the previous steps.

4.1 Data Collection and Annotation

In order to train the Machine Learning (ML) techniques used in this work, we collected images and manually annotated them to build a ground-truth corpus of map visualizations. In this section, we describe how we collected our image corpus and what aspects of the visualizations we manually annotated.

4.1.1 Image Collection

We collected our map images from three well-known geoscience journals in the field of climate change — Nature, the Journal of Climate, and Geophysical Research Letters. First, we extracted 2,018 figures from 474 documents in PDF format, using the pdffigures tool (Clark and Divvala, 2016). Then, we applied chart type classifier by Poco and Heer (Poco and Heer, 2017) to select only the map visualizations; in total, we collected 1,351 map images. We then manually applied two constraints: map images must have a color legend, and the map region must have text labels indicating the latitude and

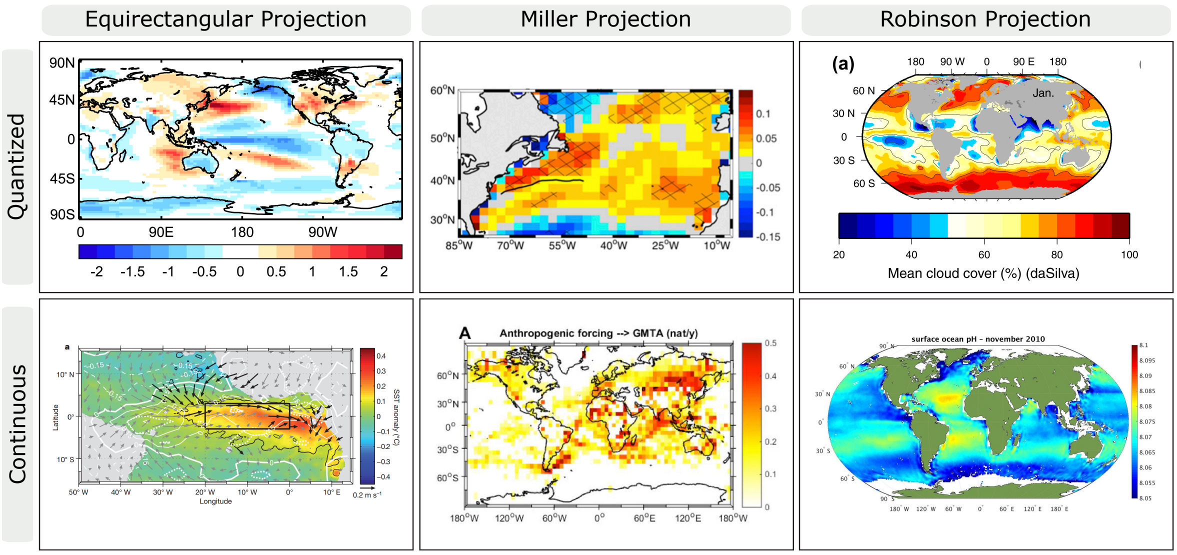


Figure 4.1: Examples of map visualization images from our corpus, covering three projections and two color legend types. We extracted map images from geoscience journals in the field of climate change.

longitude values. We leave to future work the problem of identifying a map projection from projected features only (*e.g.*, the shape and configuration of land masses).

In a preliminary analysis, we identified the projections and color legend types used on each map image. This analysis shows that the most common projections are *Equirectangular* 46%, *Robinson* 16%, and *Miller* 13% (see columns in Figure 4.1). In addition, we identified three color legend types: *discrete*, *continuous* and *quantized* (see Section 3.1). We noticed that a few percent of the collected map images contain discrete color legends. Given these results, we decided to focus on these three projections, as well as the *continuous* and *quantized* legends. Finally, we used a uniform random selection of 100 map images for each projection, including both quantized and continuous legends. Table 4.1 shows a summary of our map image corpus. Figure 4.1 shows some examples from our corpus, and we can see map images with different projections and legend types. Some images span the world, and others focus on a specific region. After selecting the 300 map images, we annotated each of them, following the process below.

	Equirectangular	Miller	Robinson	Total
Continuous	29	45	37	111
Quantized	71	50	63	189
Total	100	100	100	300

Table 4.1: Counts of map visualizations per projection and per color legend type, taken from a corpus of map images extracted from climate change publications.