# Exploring treemaps for summarizing and navigating image galleries

John A. Guerra-Gomez
Universidad de los Andes
ja.guerrag@uniandes.edu.co

Luis A. Mesa-Fajardo
Universidad de los Andes
la.mesa10@uniandes.edu.co

**ABSTRACT**

Image repositories have grown in size and meta-data, now there is the need for a tool that allows to navigate and find meaningful insights in a straightforward way. Because of this, the approach used for Treemaps result suitable for this problem. Treemaps are a visualization technique used for displaying hierarchical collections of quantitative data. In the case of images, this technique could be used to represent a large collection of pictures in a reduced space, to order images based on the pictures' meta-data and to navigate in a hierarchical way.

The thesis purpose is divided in five main stages. First, it will present an investigation of the current state of the art and how similar problems have been addressed. Then, it will be focused on present PhotoTreeMap, a system developed by Jhon A. Guerra-Gomez based on Treemaps, which aims to solve the problem of summarizing and navigating image galleries with appropriate meta-data, having in mind the best approaches of usability and visualization. Next, it will present application cases which show how the PhotoTreeMap results in a suitable solution for them. After this, the proposed solution will be evaluated through usability tests, to verify its contributions to solve the aforementioned problem. Finally, this research presents conclusions and the possibilities for future work.

## 1 INTRODUCTION

Treemaps are a two-dimensional representation of complex traditional tree structures, which show the leaf nodes as rectangular blocks with a size associated to the node's data, so the numeric value of the data will determinate the size of the block [6]. Since its creation by Ben Shneiderman in 1991 [6], the representation of image collections have followed this alternative in the recent years. It has shown how treemaps are effective in the task of provide to the user a quick overview which show the relevance of their images [1, 2, 7]. This relevance is represented by the size of the image and it could be determined by the images' metadata. For example, nowadays images are often presented along data like comments, views, likes, size, geo location, author, etc... This technique of showing images based on its metadata in a Treemap is called Photo Treemaps.

This paper will be focus in test and show the characteristics of a current JavaScript implementation of Photo Treemaps, which is developed by John A. Guerra-Gomez and called PhotoTreeMap [3]. In specific, the problem which PhotoTreemap solves, is how to visualize and navigate big image collections with rich metadata. This paper will present application usages which will demonstrate how this problem is solved and what opportunities of improvement there is in the tool.

## 2 STATE OF THE ART

The majority of systems used to navigate and summarize images repositories use a basic approach of displaying equally-sized thumbnails in a bi-dimensional grid [1]. For example, the main image explorers of operative systems like Windows or Android are a simple grid in which the user needs to scroll to navigate around the entire dataset. However, this widely used solution does not scale well and does not give the user a meaningful and intuitive interaction [2]. Because of this, other kind of systems has been developed to address this problem. Although techniques like speech recognition for annotation [4] or image processing [5] are used, this study of related work will be focus on techniques related with the use of hierarchical structures for organization of images. The above due to studies which suggest that users prefers simple navigation systems [5]. Bellow the main systems which uses this approach will be described.

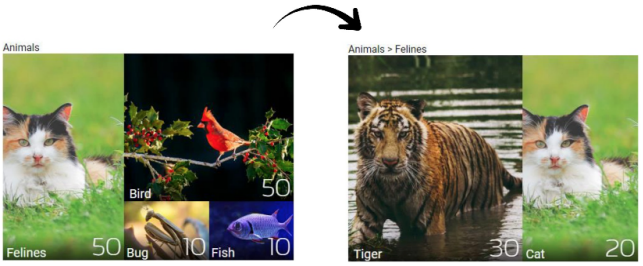


Figure 1: Animals: Basic example usage of PhotoTreeMa showing basic hierarchical structure of animals grouped by its kind. As shown above, when a group is clicked PhotoTreeMap jump in to this group and show the images of the group. [3]

First, PhotoMesa is a zoomable image browser which was the first system to use treemaps for displaying images according to directories or images' metadata. It allows the user to view multiple images in a zoomable environment, and uses a set of simple navigation mechanisms to move through the space of images [1]. In addition, it proposes two algorithms for laying out groups of images.

iMap is a treemap-based representation for visualizing and navigating image search and clustering result [7]. In general images include relevant metadata for ordering, for example, nowadays is frequent to find metadata in images related with popularity based on likes, comments or shares. Because of this, iMap suggest determinate the size of the rectangles (images) in function of its importance, such as search rank or hit count.

Browsing image datasets through Voronoi diagrams is a similar approach used with the propose of create a system in which thumbnails, whose number, shape, position, and size change dynamically and smoothly during the browsing process [2]. In this case, is evident how an appropriate spatial ordering system is used to accommodate images depending on its relevance, so the image with more focus will have more space and thus could be more detailed, on another hand, the far images from the focus does not need to represent great details so they will have smaller size.

Finally, the benefits from treemaps-oriented visualizations includes: take an optimal advantage of the available screen space, simple navigation system, proper scalability properties and the capacity to give meaningful results with respect to the entire dataset.

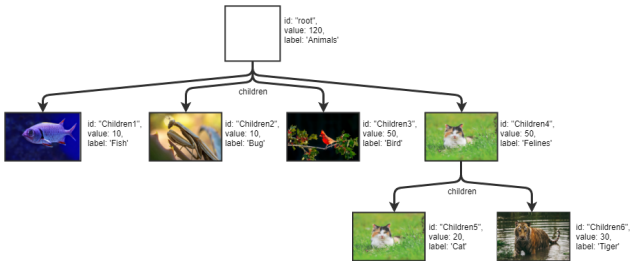


Figure 2: Hierarchical structure used for Animals. [3]

## 3 Proposed Solution

As mentioned before, the proposed solution is a visualization library based on treemaps, which have been developed by John A. Guerra-Gomez and documented and tested by Luis A. Mesa-Fajardo. PhotoTreeMap is a JavaScript library built mainly with D3, which render a Treemap of images based on a hierarchical structure. In specific, the tool receives as an input a JavaScript Object representing a Hierarchy of images grouped based in a selected attribute of the images' meta-data. Next, the hierarchy is loaded in the tool and is rendered as an output. The result is a Photo Treemap, in which the user can navigate through the different levels and leaves nodes of the hierarchy. In the next section, application cases will be described, these show the functionalities and behavior of PhotoTreeMap.

## 4 Application Cases

As mentioned before, some of the objectives of this paper is to document the library and to test the effectiveness of PhotoTreeMap. In order to accomplish this, it was necessary to create usage examples of the PhotoTreeMap which give to the users an understanding on how the tool works and how it is behavior in real contexts. In addition, they are ideal for run usability test which allows to validate the performance of the tool.

### 4.1 Animals: Application Case 1

The purpose of this example is to show and document the most basic usage of PhotoTreeMap. In this case, the example is based on a simple collection of animal images, where each image has a label, a number of likes and a category (i.e. Bug, Feline, Fish). With this information is possible to build a tree which group images based in its category. This hierarchical structure should be a JavaScript Object which have each image as a leave as shown in the Fig. 2. In order to be a valid input for PhotoTreeMap, each Node in the hierarchical structure need to have the below described properties:

- id: string, ID of the node
- value: number, Value of the node, this will be used to sort the nodes and they will be displayed depending on it.
- label: string, Text to be displayed in each Node.
- children(optional): array[Node], Each Node with children should have this property, which represent an array with nodes, which represents the node's children.
- img(optional): string, Each leave or node which represents a category must to have this property, which represents the image URL.

With this in mind, the likes will be used as the image value and the category will be used as the label. Thus, the JavaScript Object used in this example is:

```
const root = {
  id: "root",
  value: 120,
  label:'Animals',
  children: [
    {
      id: "Children1",
      value: 10,
      label:'Fish',
      img: "https://images.unsplash.com/photo-1513570050319-4797c2aef25e",
    },
    {
      id: "Children2",
      value: 10,
      label:'Bug',
      img: "https://images.unsplash.com/photo-1519167734660-d1a18d66190b",
    },
    {
      id: "Children3",
      value: 50,
      label:'Bird',
      img: "https://images.unsplash.com/photo-1482330625994-3bb3c90a5d05",
    },
    {
      id: "Children4",
      value: 50,
      label:'Felines',
      img: "https://images.unsplash.com/photo-1507984211203-76701d7bb120",
      children: [
        {
          id: "Children5",
          value: 20,
          label:'Cat',
          img: "https://images.unsplash.com/photo-1507984211203-76701d7bb120"
        },
        {
          id: "Children6",
          value: 30,
          label:'Tiger',
          img: "https://images.unsplash.com/photo-1501705388883-4ed8a543392c"
        },
      ],
    }
  ],
};
```

The result of the PhotoTreeMap for this example is a basic navigable structure which group images based on its label and order them based on its value as shown in Fig. 1. The children of the root are the first to be displayed, in this case the Felines has a total of 50 likes, thus it is the node which occupies more area in the PhotoTreeMap. In another hand, the category Fish is the smaller node because to it is the node with less likes. One of the key characteristics of the PhotoTreeMap is the possibility of jump between groups in the tree. For example, when a Node with children is clicked the PhotoTreeMap zoom-in to show those children. The purpose of this example is to document and show to the library users the most basic to use the PhotoTreeMap. In conclusion, this example allows the user to have a straightforward way of find the category which have more likes and allows him to navigate through the different groups of the dataset in a simple way.

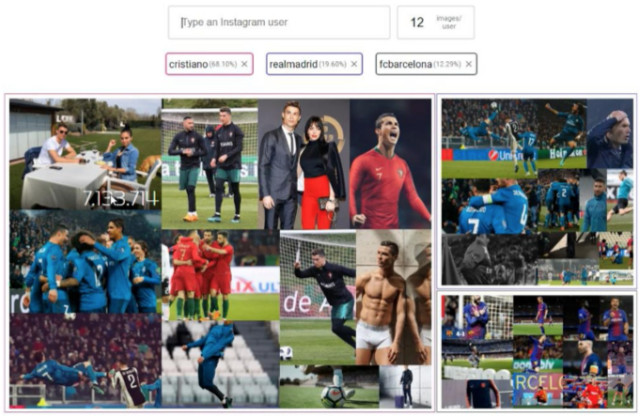


Figure 3: Instagram Trends: PhotoTreeMap used to compare the popularity of Instagram users [3]

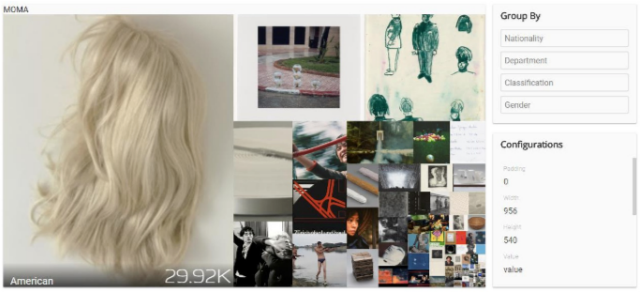


Figure 4: MOMA Navigator: PhotoTreeMap used to navigate around the artworks of The Museum of Modern Art (MOMA) based in its characteristics. [3]

### 4.2 Instagram Trends: Application Case 2

Instagram Trends create a PhotoTreeMap to compare the popularity of Instagram users. This example requires a medium configuration and follow the same idea as the before example, the size of the images in the TreeMap are determinated by the relevance of each image. In specific, Instagram Treends use a PhotoTreeMap to display any user in Instragram based on its popularity, where the popularity is measured by the likes in their photos. So, when a user types an Instagram user, the application request the most recent pictures of the user to the Instagram API. After this, a hierarchical structure is created with the old and the new photos of the users and the PhotoTreeMaps is updated. One key difference with respect to the Animals example, is that the groups of photos are not presented like zoomable structure where the user can jump between levels, in another hand it represent all the images at the same time as Fig. 4 shows. It is a feature that can be configured in PhotoTreeMap. Finally, this implementation has shown a practical use which allows the user to find the most relevant user in Instagram in a straightforward way.

Due to recent changes in the privacy policies of Instagram this example is not longer working. However, documentation, source code and a demo video of it could be founded at the PhotoTreeMap repository [3].

### 4.3 Moma Navigator: Application Case 3

MOMA Navigator create a PhotoTreeMap to navigate around the artworks of The Museum of Modern Art (MOMA) based on its characteristics. This example require a high configuration and load 131000 records representing the MOMA's artworks. This is the most complex example and it was used to evaluate the effectiveness of the PhotoTreeMap for a considerable amount of data and with a high complexity in its hierarchical structure. This application have two new components which aims to explain to the user all the possible configurations available in the PhotoTreeMap and allow him to create different hierarchical structures with the data. In this case, the group criteria could be defined by the user interactively, the user could select from the interface in which order want to group the artworks. For example they could be grouped by Nationality, then Department, then Classification and then Author's genre. In addition, the value of the nodes is determined by the amount of its children, so the size of the Node represents how big is this group. Finally, this implementation has shown effectiveness to navigate around the MOMA artworks and it could be now tested against the official MOMA explorer present in its web page.

## 5 Evaluation

In order to verify the performance of the tool it was necessary to run usability tests and get feedback from the general public. The tests were executed over the MOMA Navigator, because it was the example with more features and data to explore. The usability test was focus on request the users to complete the following tasks:

- Which department have the most art works?
- What is the American gender which have the most art works?
- Which country have contributed the most to the Photography department?
- Which classification have more art works in the Photography department?
- Describe an art work, which belong to the Films department, with a classification of Video and made by an Austrian women.
- Describe an art work, which belong to the Films department, with a classification of Paint and made by an American man.

After the users complete theses tasks the following questions were asked:

- Who could use this tool?
- There was something that block you to complete the tasks?
- Q1: How useful do you think is this tool for finding the subcategory with more images? (1 - Not useful , 5 - Really useful)
- Q2: How useful do you think is this tool for finding the number of images in a specific subcategory? (1 - Not useful , 5 - Really useful)
- Q3: How easy do you think it feels to use the tool? (1 - Really hard , 5 - Really easy)
- Q4: Compared with other tools of image management that you know or have used, how do you rate the utility of the PhotoTreeMap? (1 - It is less useful than the others , 5 - It is more useful that the others)
- Any suggestions?

Six users completed the usability test with an average good result. Almost all user had problems at the beginning of the test, because this was the first time that they use a tool like this. However, once they completed the first task all the following tasks were relatively easy to accomplish for them. As a general conclusion the users said that some visual components should be improved, for example they disagree in the current behavior of the labels, because they just show up when the mouse is over the node, so it difficult the task of find

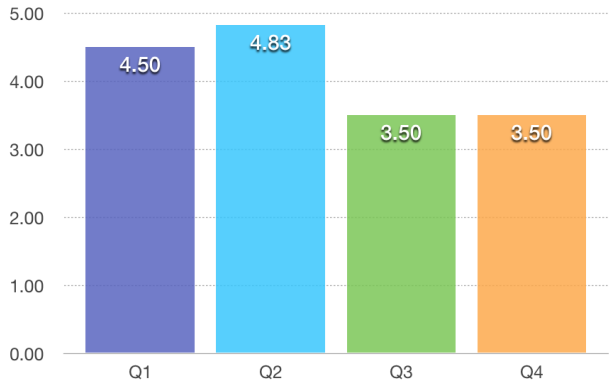


Figure 5: Users rates by question

a specific node. In addition, they suggest to pay attention to the font size, because smaller nodes have an almost unreadable font. They also suggest too to improve the right component in charge of reordering the structure, because this is confusing for them and could requires more instructions. Finally, the users tended to rate positively the PhotoTreeMap as shown in Fig. 5 and add valuable comments about the future development of the tool, for example they told us how fields like arts, design or music could use this tool.

## 6 Conclusion

As mentioned before, the paper purpose was to validate that the PhotoTreeMap is a suitable solution for summarizing and navigating image galleries with appropriate meta-data. This affirmation is validated by the results of the usability tests, which show how users find the tool useful to navigate through image collection with appropriate meta-data. However, in order to have more reliable results the tests should be executed over a wide range of users. In addition, as described before there is the need to improve the tool based on the feedback given by the users.

## References

[1] B. B. Bederson. PhotoMesa: A Zoomable Image Browser Using Quantum Treemaps and Bubblemaps.

[2] P. Brivio, M. Tarini, and P. Cignoni. Browsing large image datasets through Voronoi diagrams. *IEEE Transactions on Visualization and Computer Graphics*, 2010. doi: 10.1109/TVCG.2010.136

[3] John Guerra, Luis Mesa. PhotoTreeMap. `https://github.com/john-guerra/photoTreemap/`, 2018. [Online; accessed 01-Jun-2018].

[4] T. J. Mills, D. Pye, D. Sinclair, and K. R. Wood. SHOEBOX: A DIGITAL PHOTO MANAGEMENT SYSTEM.

[5] S. Reddy, A. Parker, J. Hyman, J. Burke, D. Estrin, and M. Hansen. Image Browsing, Processing, and Clustering for Participatory Sensing: Lessons From a DietSense Prototype.

[6] B. Shneiderman. Tree visualization with tree-maps: 2-d space-filling approach. *ACM Transactions on Graphics*, 1992. doi: 10.1145/102377.115768

[7] C. Wang, J. P. Reese, H. Zhang, J. Tao, and R. J. Nemiroff. iMap: A stable layout for navigating large image collections with embedded search.