Research Report:
Saliency-Assisted Navigation of Very Large Landscape Images

Brian Brubach, Mohamed K. Gunady, Eric Krokos, Peter Sutor
Original Paper By: Cheuk Yiu Ip and Amitabh Varshney
University of Maryland - College Park

Introduction

Imagine a group of hikers lost in a park with no means of communication or signaling. Traditional search and rescue techniques require many man-hours and often turn up nothing. So the ability to detect the presence of interesting objects within a current high-resolution image of the park would be highly desirable in such a scenario.

Unfortunately, human ability to capture large amounts of data, namely photographs, has outpaced our ability to process them. These large photographs, referred to as gigapixel images, are generated by taking many pictures of a scene and stitching them together. Consider a four-gigapixel image displayed on a two-megapixel monitor. Every monitor pixel would represent 2,000 image pixels, showing only 0.05% of the overall data. There are many systems that allow a user to zoom in and navigate these images, but it can still be difficult and time-consuming to find interesting regions within these images. Prof. Varshney and his GVIL lab [3] at UMD provide an efficient, automatic saliency detection algorithm for the purpose of finding the interesting parts of a large image and then making navigation easier. A video overview of their approach can be found at [2].

Example of identifying interesting objects within the scene, then zooming in to appropriate scale.

In addition to the aforementioned rescue scenario, this work has many far-reaching applications. Manufacturers of semiconductor wafers use terapixel images for quality inspection and astronomers gather vast amounts of such images to explore the universe. The speed of this algorithm, can allow military and police forces to process high-resolution images in a reasonable timeframe to locate individuals or objects. The medical field can quickly find salient areas in large cross-sectional imagery to aid in the identification of health problems. The days of spending many man-hours manually searching and navigating enormous images are over.
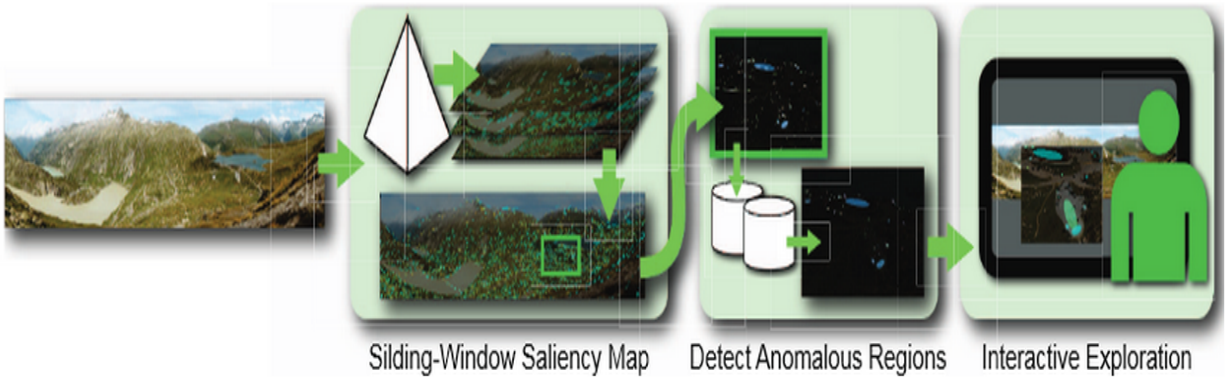
Steps in the system. The first step builds a saliency map by adding a sliding window with scales to the traditional multi-scale image saliency approach. The second steps identifies any unique regions with color descriptors. Lastly, users eliminate false positives.

What This Work Entails

The problem of exploring very high-resolution landscape images introduces three challenges that Ip et al. [1] target:

1. Visually communicating massive images that exceed the capabilities of the human visual system and present-day display software and hardware.

2. Designing effective algorithms that identify informative regions hidden among the vast majority of the unimportant data within a very large image.

3. Processing large amounts of data while preserving reasonable performance, even for routine image-processing operations.

The work of Ip et al. is the first step towards addressing these challenges. Consequently, it presents an extension of classical algorithms for image saliency to very large images. They also present an interactive visual exploration system that balances between automatic extraction algorithms and user preferences.

How The Algorithm Works

The algorithm used in this system takes advantage of the parallel power of the GPU, or graphics processing unit. Given the size of these images, the primary bottleneck of this and similar algorithms is the movement of data from disk to GPU memory.

Once the data is loaded into memory, the first step performed is to build a saliency map. This is done by a highly parallelizable approach using a sliding window over the entire image and looking at two primary features — intensity and color. This sliding window is applied at several different scales since some objects only appear salient at a certain scale. For instance, a building may not stand out at a fine scale, yet appear very contrasting to its surroundings at a coarse scale.

After the saliency map is constructed, the algorithm removes repeating elements which may not be as salient. To accomplish this, it relies on color descriptors to identify the most unique regions using a k-Nearest Neighbor anomaly detection.

Finally, the remaining regions are used as way-points in an interactive fly-through, allowing the user to focus on suspected anomalies instead of meticulously scanning through
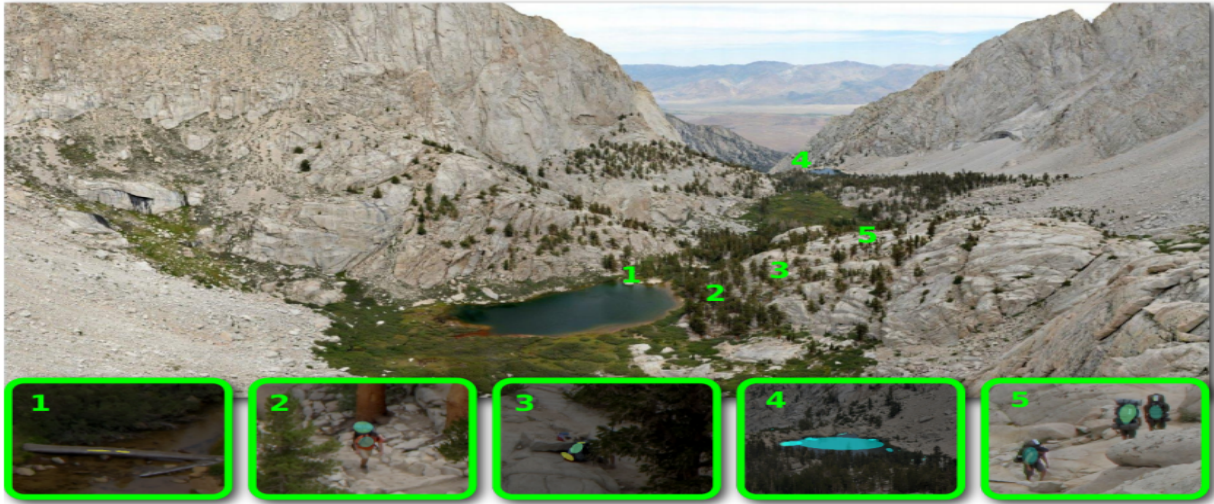
A five-gigapixel image of Mt. Whitney. Detected regions are shown by number. The algorithm was able to detect 11 out of 15 of the hikers in this image.

the entire image. In this step, the user can delete remaining false positives identified in the earlier phases of the algorithm and a batch delete option is provided to further minimize time and workload. The user is able to finely specify a type of misidentified region and all sufficiently similar regions are removed.

A Brief Demonstration

One example Ip et al. used to test their algorithm is a five-gigapixel view of Mt. Whitney. This is an image of a rocky valley with trees and other natural flora. Without providing a person detector, the algorithm was able to detect 11 of 15 hikers present in the image. It also detected anomalous natural features. The most interesting is a tree that has fallen onto its side over a stream. This makes sense considering most trees are upright and over land. Overall, it took the system 11 hours to process the image, with an average of 2.5 hours per gigapixel. However, as stated before, the vast majority of the time was spent on the movement of data.

References

[1] Ip, Cheuk Yiu, and Amitabh Varshney. "Saliency-assisted navigation of very large landscape images." *Visualization and Computer Graphics, IEEE Transactions on 17.12* (2011): 1737-1746.

[2] Youtube Video: https://www.youtube.com/watch?v=FwSMjYHTNX8

[3] GVIL research lab: http://www.cs.umd.edu/ varshney/

Email Addresses

- Brian Brubach: bbrubach@umd.edu

- Mohamed Gunady: mgunady@cs.umd.edu

- Eric Krokos: SuperSmashcz@yahoo.com

- Peter Sutor Jr: psutor@umd.edu

Written on February 26th, 2015.