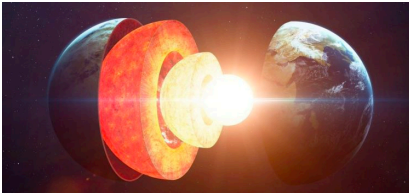
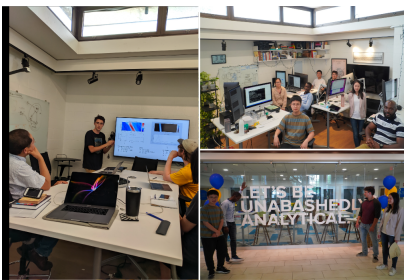


Talent Hiring RA, Spring '24

Earth Imaging Lab



<People | Ideas | Outreach>



Above: Our Lab. Feel free to set up a meeting to discuss your progress with the projects.

Office Hours:

M-F (3-4 PM) @ Hutchison 329



EI.zip

Interview Warmup

UPDATED: 28, FEB '24

3 Projects – 1 Position

Summary

Thank you for your interest in joining the Earth Imaging Lab. We are impressed with your resume. To help us understand your preparedness to fill this position, we have designed three simple projects that can be completed in a few days. You will write code but also think analytically. We are not just looking for a final solution, we want to see how you think, solve problems, and work in a team. We will send you an assigned project based on your ranking of the project that appeals to you. The applicants with the best solutions will be selected for follow-up interviews. See below for details on the problem statements. To see the data and starter codes use QR code or link: <http://tinyurl.com/33zs99en> . For extra credit, solve all three tasks.

Project Descriptions (problem, code and data)

1. **Prj_NetworkXX (Graphs):** You have two files: `node_locations.csv` and `node_alive.csv`. The nodes represent stations located in and around Africa. Their locations are found in the first file, and the dates of operation are found in the second. The goal is to produce two sorted lists: `node_dates` and `node_pairs`. The first list should include nodes, starting with nodes with most connections, and include only dates overlapping with other nodes. The second list should include all connected station-pairs and the dates of common operation, sorted by pairs with the most connections. Hint: use bipartite and weighted graphs.
2. **Prj_GPU (Fast Algorithms):** You have three directories and two scripts. The scripts are in the `CODE` dir: `stp1_initparams` and `stp2_doCCF`. The scripts compute pairwise station correlations using ground vibration data obtained at stations and stored in the `DATA` dir as `*SAC` files. Results are saved in the `RESULTS` dir. The goal is to time this code and design and implement a GPU algorithm that speeds up the current computation with benchmarks that demonstrate improvements. Hint: reduce loops, streamline matrix operation, parallelize computations, reduce memory operations.
3. **Prj_GAN (Generative AI):** You have been given a simulated dataset and a pre-trained generative model (see `readme.txt`). The training data is in `images.csv` and `coords.csv`. The generative model is in `WGAN.model`. Your first task is to use the pre-trained model to generate a new image. Your second task is to re-train your own WGAN model using a suitable optimization strategy. For the second task, describe statistics for benchmarking your generative model.

Submission Protocol

1. Submit solution as **python notebooks or matlab livescripts** to <https://repovibranium.quickconnect.to/sharing/Ws7vh96g6>