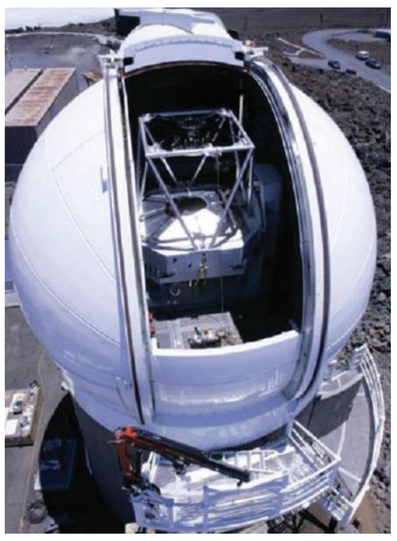
Indexing a Large-Scale Database of Astronomical Objects

Bin Fu, Eugene Fink, Garth Gibson, Jaime Carbonell
Carnegie Mellon University

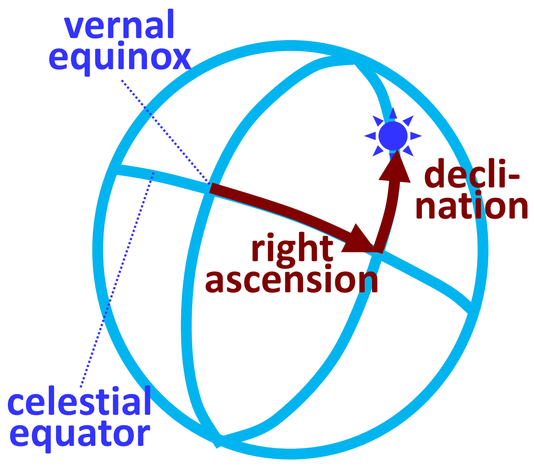


When astronomers analyze telescope images, they match the observed objects to the catalog. Since the objects may "move" slightly from image to image because of atmospheric and optical distortions, astronomers need to retrieve close approximate matches. We have developed a matching procedure that maintains a catalog with billions of objects and processes millions of observed objects per second.

Problem

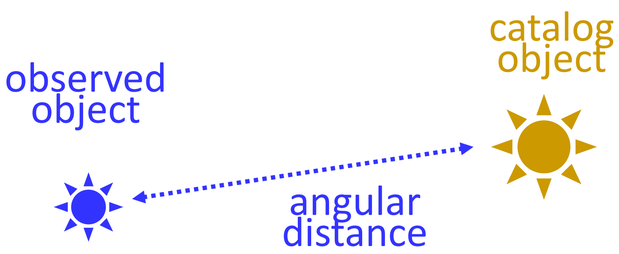
We describe a celestial object by three values:

- **Declination** (“latitude”): Angle from the celestial equator
- **Right ascension** (“longitude”): Angle from the vernal equinox
- **Magnitude**: Brightness on a logarithmic scale



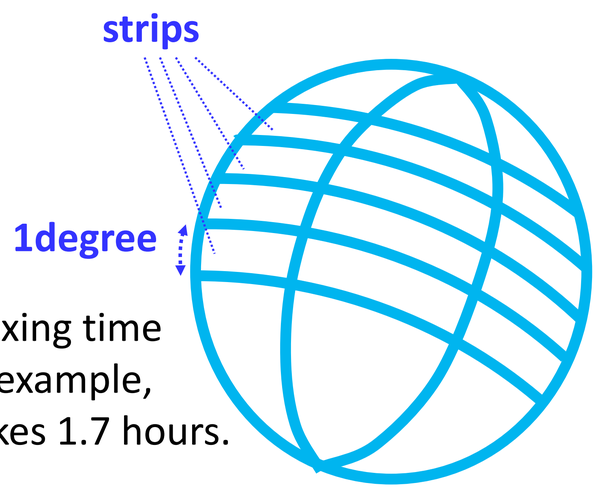
When astronomers get a new sky image, they compare it with the catalog and identify previously unseen objects, such as supernovae and asteroids. An image covers a small rectangular region, and the astronomers match all its objects against the catalog.

- The similarity of an observed object and its potential catalog match is a function of the angular distance and the magnitude difference.
- If the distance is greater than an arcsecond (1/3600 degrees), there is no match.
- For each object, we need to identify the closest match (if any).



Indexing

- **Buckets by declination**: Divide the catalog objects into "buckets" by declination, where each bucket corresponds to a horizontal strip; the width of each strip is 1 degree, and the number of strips is 180.
- **Sorting by right ascension**: For each strip, sort the objects within the respective bucket by the right ascension.
- Store the resulting 180 sorted arrays on disk.

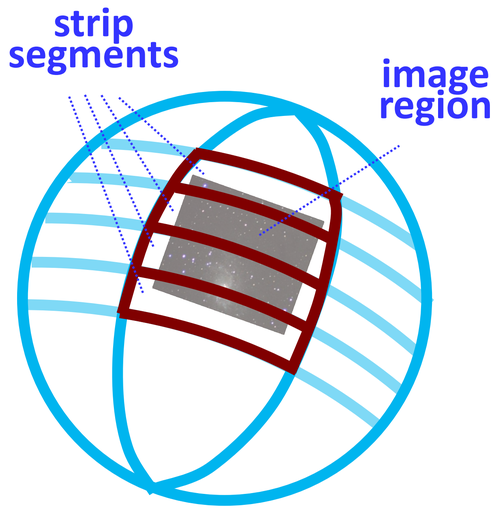


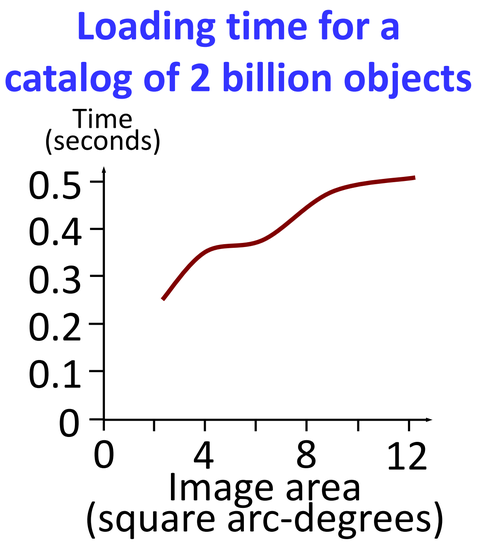
For a catalog of *n* objects, the indexing time is 0.1 * *n* * lg *n* microseconds. For example, the indexing of 2 billion objects takes 1.7 hours.

Retrieval

Loading:

- Load the list of observed objects from an image and determine the rectangular region that contains them.
- Identify the "strip segments" of the catalog that overlap that region.
- Load these segments into memory.





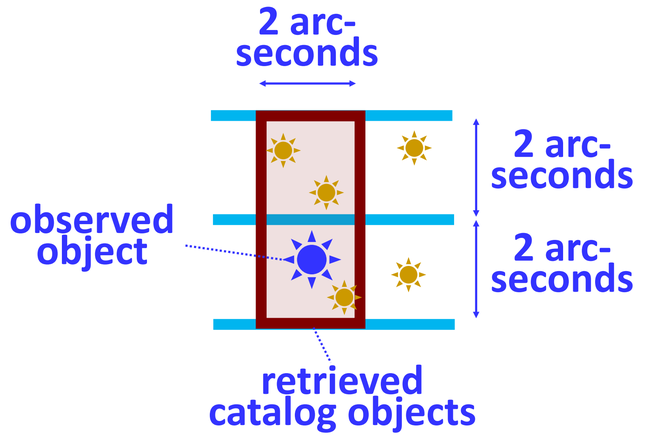
Splitting:

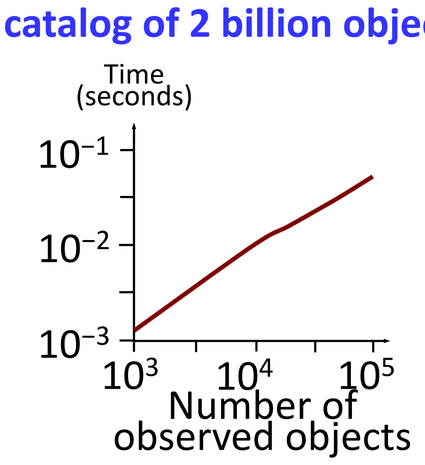
- Split the strips of the loaded region into substrips; the width of each substrip is 2 arcseconds (1/1800 degrees.
- For each substrip, sort its objects by the right ascension

Matching:

For each observed object:

- Identify the substrip that contains it and the closest substrip that does not contain it.
- For these two substrips, retrieve all catalog objects that are within an arcsecond from the observed object.
- Loop through these objects and identify the closest match (if any).





Distributed Technique

- Process different images in parallel.
- If the overall memory is sufficient for storing the entire catalog, avoid the loading step.

Carnegie Mellon

Parallel Data Laboratory