catsHTM: A Tool for Fast Accessing and Cross-matching Large Astronomical Catalogs

Maayane T. Soumagnac and Eran O. Ofek
Benoziyo Center for Astrophysics, Weizmann Institute of Science, 76100 Rehovot, Israel; maayane.soumagnac@weizmann.ac.il
Received 2018 April 16; accepted 2018 May 9; published 2018 June 12

Abstract

Fast access to large catalogs is required for some astronomical applications. Here we introduce the catsHTM tool, consisting of several large catalogs reformatted into HDF5-based file format, which can be downloaded and used locally. To allow fast access, the catalogs are partitioned into hierarchical triangular meshes and stored in HDF5 files. Several tools are provided to perform efficient cone searches at resolutions spanning from a few arc-seconds to degrees, within a few milliseconds time. The first released version includes the following catalogs (by alphabetical order): 2MASS, 2MASS extended sources, AKARI, APASS, Cosmos, DECaLS/DR5, FIRST, GAIA/DR1, GAIA/DR2, GALEX/DR6Plus7, HSC/v2, IPHAS/DR2, NED redshifts, NVSS, Pan-STARRS1/DR1, PTF photometric catalog, ROSAT faint source, SDSS sources, SDSS/DR14 spectroscopy, SkyMapper, Spitzer/SAGE, Spitzer/IRAC galactic center, UCAC4, UKIDSS/DR10, VST/ATLAS/DR3, VST/KiDS/DR3, WISE and XMM. We provide Python code that allows to perform cone searches, as well as MATLAB code for performing cone searches, catalog cross-matching, general searches, as well as load and create these catalogs.

Key words: catalogs – methods: data analysis – surveys

Online material: color figures

1. Introduction

In the past three decades, the emergence of catalog services like SIMBAD (Wenger et al. 2000), VizieR (Ochsenbein et al. 2000), the NASA Extragalactic Database (NED) and MAST, have had, and continue to have, an enormous impact on astronomical research. These services are being used extensively by the astrophysics community and are used in a large fraction of articles. Nevertheless, one obvious limitation of all these services is that the Internet connection limits the speed of search. This is an obstacle for some applications requiring very fast access to large catalogs. Examples of such applications are cross-matching multiple large catalogs (e.g., Malkov et al. 2012; Ogle et al. 2015), and vetting transient candidates detected by synoptic surveys. For example, surveys like Pan-STARRS (Chambers et al. 2016) and the Palomar Transient Factory (Law et al. 2009) generated a large number of transient candidates which have to be cross-matched with multiple catalogs in order to be vetted, classified and followed up.

Cross-matching tools developed in the last years include the web-based tools CDS-Xmatch (Pineau et al. 2011), ARCHES (Motch & Arches Consortium 2015; Pineau et al. 2017) and SkyQuery (Budavári & Szalay 2008; Budavári et al. 2013) or local command line tools like TOPCAT (Taylor 2005), STILTS (Taylor 2006) and C (Riccio et al. 2017) which allow to overcome some of the speed limitations of the web-based applications. Another simple solution is to use local relational databases. Indeed, with proper indexing (e.g., Hierarchical Triangular Mesh (HTM); Szalay et al. 2007), Structured Query Language (SQL) queries on such database are relatively fast. In this paper we present a simple and—at least from our experience—faster alternative.

We provide a set of large catalogs stored in HDF5 files. HDF5 is a data model, library and file format for storing and managing data. It supports an unlimited variety of data types and is designed for flexible and efficient I/O and for high volume and complex data. Furthermore, HDF5 tools are available in many computer languages. The data storage methodology we use is designed to provide good performances both for small size (i.e., a few arcsec) and large size (i.e., deg) cone searches. In addition to the formatted catalogs, we provide a set of tools to perform fast cone search, serial search, catalog cross-matching and catalog generation.

The structure of this paper is as follows. In Section 2, we detail the structure and format of the catalogs. In Section 3 we list all the catalogs currently available and present the codes

---

1. http://simbad.u-strasbg.fr/simbad
2. http://vizier.u-strasbg.fr
3. https://ned.ipac.caltech.edu
4. https://archive.stsci.edu/astro
5. https://support.hdfgroup.org/HDF5/
we provide for fast access to these catalogs. We compare our tool to other existing tools and discuss our results in Section 4.

2. The Data Format

The efficiency of catsHTM lies in three aspects: (1) the way the data is partitioned into files; (2) the way the data is stored in these files; and (3) the data indexing.

The data of each catalog is partitioned in the following way. We divide the celestial sphere using a Hierarchical Triangular Mesh (HTM) quad-tree (Kunszt et al. 2000; Szalay et al. 2007; Budavári et al. 2010). This method is based on a recursive subdivision of the celestial sphere into spherical triangles of similar shapes, called trixels. The HTM method of dividing the sphere is particularly good at supporting searches at a wide range of resolutions (hemispheres to arc seconds). In our case, depending on the size of the catalog, the level of the HTM (i.e., the number of subdivisions necessary to create the smallest trixel) varies between six and nine. The amount of levels we chose provides good efficiency for cone searches at resolutions spanning from a few arc-seconds to a degree—i.e., the typical resolutions usually used for astronomical applications. Each trixel in the quad-tree is allocated a number, which is referred to as the trixel index throughout this paper. In Figure 1, we show an illustration of the recursive decomposition of a sphere into HTM trixels.

To store the data of the partitioned catalogs, we use data sets, one of the object types available within the HDF5 data model (see footnote 5). A data set is simply a multidimensional array and a HDF5 file can contain multiple data sets.

For each trixel, we create a pair of data sets:

1. The trixel-data set, containing the actual catalog data that are within the trixel. Only highest level trixels are populated (and exist). The data in each trixel is sorted by declination.

2. The index-data set, containing a two column matrix. The first column is the line number in the trixel-data set and the second column is the corresponding declination. This data exist in steps ranging from 30 to 300 lines and can be used in some cases to expedite a search within a trixel.

This is mainly used for catalogs cross-matching.

These pairs of data sets are then stored, in groups of 100 (by default), in HDF5 files.

For each partitioned catalog, we create an additional HDF5 file, called the HDF5 index file, which contains the information needed in order to access the relevant trixels, when making a cone search. The HDF5 index file contains as many lines as trixels in the tree. Each line corresponds to one trixel and contains: the index of the trixel; the indexes of the parents and sons trixels; the sky coordinates (latitude and longitude) of the poles of the great circles that defines the trixel; the number of sources it contains (NaN if the trixel does not belong to the highest level in the tree). The HDF5 index file allows to perform an efficient tree search for all the trixels that intersect—or are contained within—a given cone. Finally, we also keep a file containing additional meta data, such as the catalog column names and units.

To summarize: each catalog is partitioned into trixels. For each partitioned catalog, the following files exist: (1) HDF5 files containing data sets where the catalog data can be stored; (2) a HDF5 file called the index file, where the index and sky coordinates of each trixel are stored; and (3) a file containing meta data on the catalog.

In Table 1, we present the default files and data sets naming formats we used. Additional documentation and examples are available online.7

3. Available Catalogs and Codes

The catalogs currently available are listed in Table 2 and the list will be updated regularly in the online documentation (see footnote 7). The catsHTM directory containing the HTM/HDF5 catalogs requires ~2.1 TB of disk space. Download instructions as well as links to the code used to create those catalogs can be found in the online documentation (see footnote 7).

---

5 A spherical triangle is a polygone of which the edges are segments of three great circles.

7 https://webhome.weizmann.ac.il/home/eofek/matlab/doc/catsHTM.html
| Object | Default naming format | type   |
|--------|----------------------|--------|
| Trixel-data set | htm_%06d | HDF5 data set |
| Index-data set | htm_%06d_Ind | HDF5 data set |
| HDF5 files containing the trixel-data sets and the index-data sets | (CatBaseName)_htm_%06d.hdf5 | HDF5 file |
| HDF5 index file | (CatBaseName)_htm_Ind.hdf5 | HDF5 file |
| Catalog metadata (e.g., column names) | (CatBaseName)_htmColCell.mat | .mat file |

Note. "(CatBaseName)" is the catalog name (see Table 2). The default number of pairs of trixel- and index-data sets in each HDF5 file is 100. In the HDF5 index file, the catalog file name is rounded such that the last two significant digits are always 0 (in case of 100 trixels per HDF5 file).

| CatBaseName | Catalog | \( N_{\text{src}} \) | \( N_{\text{col}} \) | Reference |
|------------|---------|----------------|----------------|----------|
| TMASS      | 2MASS   | 470 992 608    | 10             | Skrutskie et al. (2006) |
| TMASSxsc   | 2MASS extended sources | 1 647 599 | 15 | Jarrett et al. (2000) |
| AKARI      | AKARI   | 870 973        | 9              | Ishihara et al. (2010) |
| APASS\(^a\) | APASS   | 55 395 532     | 19             | Henden et al. (2009) |
| Cosmos     | Cosmos  | 438 226        | 27             | Capak et al. (2007) |
| DECaLS    | DECaLS/DR5 | 679 250 688 | 26 | http://legacySurvey.org/decamls/ |
| FIRST      | FIRST   | 946 432        | 14             | Becker et al. (1995) |
| GAIADR1    | GAIADR1 | 114 268 060    | 8              | Gaia Collaborationi et al. (2016) |
| GAIADR2    | GAIADR2 | 1 692 967 552  | 27            | Gaia Collaboration et al. (2018) |
| GALEX      | GALEX/DR6Plus7 | 165 794 048 | 9 | Martin et al. (2005) |
| HSCv2      | Hubble Source Catalog HSC/v2 | 318 758 784 | 15 | Whitmore et al. (2016) |
| IPHAS      | IPHAS/DR2 | 205 773 248 | 17 | Barentsen et al. (2014) |
| IRACgc     | Spitzer/IRACgc | 1 065 565 | 15 | Ramírez et al. (2008) |
| NEDz       | NED redshifts (2018 May 2 version) | 7 154 168 | 8 | Helou et al. (1990); Mazzarella & NED Team (2017) |
| NVSS       | NVSS    | 1 773 486      | 12             | Condon et al. (1998) |
| PS1\(^b\)  | Pan-STARRS1/DR1 | 2 612 435 712 | 41 | Chambers et al. (2016) |
| PTFpc      | PTF photometric catalog | 21 167 678 | 14 | Ofek et al. (2012) |
| ROSATfse   | ROSAT faint source | 105 924 | 21 | Voges et al. (2000) |
| SAGE       | Spitzer/SAGE | 9 094 829 | 16 | Meixner et al. (2006) |
| SDSSDR10   | SDSS sources | 447 279 840 | 16 | Ahn et al. (2014) |
| SkyMapper  | SkyMapper DR1.1 | 285 159 328 | 30 | Wolf et al. (2018) |
| SpexSDSS   | SDSS/DR14 spectroscopy | 4 311 570 | 32 | Abolfathi et al. (2017) |
| UCAC4      | UCAC4   | 113 780 216   | 45             | Zacharias et al. (2013) |
| UKIDSS     | UKIDSS/DR10 | 79 333 520 | 38 | Lawrence et al. (2007) |
| VSTatlas   | VST/ATLAS/DR3 | 106 556 464 | 14 | Shanks et al. (2015) |
| VSTkids    | VST/KIDS/DR3 | 48 735 816 | 21 | de Jong et al. (2017) |
| WISE       | WISE    | 563 908 224   | 37             | Wright et al. (2010) |
| XMM        | XMM     | 727 790        | 14             | Rosen et al. (2016) |

Notes. List of the catalogs available at the date of submission (this list will be regularly updated in the online documentation). In some cases, there are small discrepancies in the number of sources (typically \(<10^{-5}\) between the version available on VizieR and our HDF5 version. Some of these differences are due to problems in the ingestion process and this will be fixed in future releases.

\(^a\) A newer version of the APASS catalog will be included in one of the next releases.

\(^b\) Sources detected in the stacked images above a certain signal-to-noise ratio; the current version of the HDF5 catalog has some missing regions below \(\delta = 0\).

We provide two sets of codes to access the formatted catalogs listed in Table 2:

1. A Python code\(^8\) that provide cone search functions;

2. A MATLAB code\(^9\) that can perform a variety of tasks in addition to cone search, described in the online documentation (see footnote 7): general searches, cross-

---

\(^8\) https://github.com/maayane/catsHTM

\(^9\) https://webhome.weizmann.ac.il/home/eofek/matlab/doc/install.html
matching, plotting, loading catalogs and generating new catalogs. For example, in Figure 2 we show a sky map, generated by one of these functions, that presents the source density in the SDSS catalog.

Currently, the Python code is compatible with both Python 2 (higher than 2.7.10) and Python 3 and requires a small amount of standard basic packages (numpy, scipy, math and h5py). The MATLAB code is available as part of the MATLAB Astronomy & Astrophysics Toolbox10 (Ofek 2014), and has been tested with MATLAB R2016b. These requirements will be updated, if needed, in the online documentation, as future versions of catsHTM are released.

4. Results and Discussion

Compared to one specific relational database SQL searches, we find our code to be about one order of magnitude faster for cone searches. A 10° radius cone search, on a random celestial position, using a Xeon(R) CPU E5-2670 v3 at 2.30 GHz machine takes on average about 1–4 ms using the MATLAB tool and 1–40 ms using the Python tool. A 1000° radius cone search on the same machine takes on average 4–300 ms using the MATLAB tool and 2–600 ms using the Python tool, where the exact time depends on the catalog.

Our code can also be used to efficiently cross-match catalogs and perform general searches. These options can use multiple processors. The cross-matching is performed by loading a single trixel data set from the first catalog into memory and cross-matching it only with sources that are found in overlapping trixels from the second catalog. For example, using 24 processors, cross-matching of the APASS catalog against itself takes about 160 s while cross-matching the 2MASS catalog against the WISE catalog takes about 53 minutes (without dumping the results). The cross-matching tool is work in progress. In particular, it is currently only available in MATLAB and will be available in Python in future releases of catsHTM. In the future, we plan to extend the format for multiple-epoch catalogs, to add new catalogs and to provide catalogs of cross-matchings between all the catalogs.

Making a fair comparison between the performances of catsHTM and those of other available tools is a complex task, because catsHTM comes with formatted catalogs. Web-based tools like OpenSkyQuery (Nieto-Santisteban et al. 2006) and CDS-Xmatch (Pineau et al. 2011), as well as tools with a strong graphical component such as TOPCAT (Taylor 2005) consist of portals which have the disadvantage of being relatively hermetic to the user. The need to allow access to multiple users comes with its own set of compromises: CDS-Xmatch limits both the disk space and computation time available to the users.

On the contrary, catsHTM has been designed as a stand-alone tool, to avoid the above disadvantages and allow high flexibility. The simple cone search python and MATLAB codes can be easily edited according to the user’s specific needs and easily integrated in the user’s own pipelines. The all-in-one structure of the code and catalogs allows the user to run jobs on his own computer and without relying on internet connection. The C3 (Riccio et al. 2017) and Nway (Salvato et al. 2018) tools have been designed in this spirit, but focus on cross-matching, whereas the primary task of catsHTM is to perform efficient cone-search queries.

Perhaps the existing tool closest to what catsHTM is trying to achieve is STILTS (Taylor 2006) which, on the one hand, is a powerful stand-alone tool, free of the disadvantages of webpage applications, and on the other hand offers a large variety of catalog-handling functionalities, including cone search.

catsHTM is unique in the sense that it is not limited to the search and cross-matching tools it offers. These tools come together with a large—and constantly updated—set of astronomical catalogs, formatted into the binary HDF5 format.

E.O.O. is grateful for the support by grants from the Israel Science Foundation, Minerva, Israeli Ministry of Science, the US-Israel Binational Science Foundation, and the I-CORE Program of the Planning and Budgeting Committee and The Israel Science Foundation.

M.T.S. acknowledges support by a grant from IMOS/ISA, the Ilan Ramon fellowship from the Israel Ministry of Science and Technology and the Benoziyo center for Astrophysics at the Weizmann Institute of Science. M.T.S. would also like to thank Steve Schulze, Barak Zakay, Ronen Tamari and Adam Rubin for useful discussions.

References

Abolfathi, B., Aguado, D. S., Aguilar, G., et al. 2017, arXiv:1707.09322
Ahn, C. P., Alexandroff, R., Allende Prieto, C., et al. 2014, ApJS, 211, 17
Barentsen, G., Farnhill, H. J., Drew, J. E., et al. 2014, MNRAS, 444, 3230
Becker, R. H., White, R. L., & Helfand, D. J. 1995, ApJ, 450, 559

---

Figure 2. Source density in the SDSS-DR10 catalog as plotted using the catsHTM.plot_density function.
(A color version of this figure is available in the online journal.)

10 https://webhome.weizmann.ac.il/home/eofek/matlab
