

## DEVELOPMENT OF A DATABASE FOR BENCHMARK DATASETS IN PHOTOGRAMMETRY AND REMOTE SENSING

Lina E. Budde<sup>1,\*</sup>, Jakob Schmidt<sup>1</sup>, Arash Javanmard-Ghareshiran<sup>1</sup>, Sebastian Hunger<sup>1</sup>, Dorota Iwaszczuk<sup>1</sup>

<sup>1</sup> Technical University of Darmstadt, Dept. of Civil and Environmental Engineering Sciences,  
Remote Sensing and Image Analysis, Darmstadt, Germany  
(lina.budde, jakob.schmidt1, arash.javanmard-ghareshiran, sebastian.hunger, dorota.iwaszczuk)@tu-darmstadt.de

### Commission I, WG I/10

**KEY WORDS:** Benchmark, Database, Dataset, Metadata, Photogrammetry, Remote Sensing.

### ABSTRACT:

Data are a key component for many applications and methods in the domain of photogrammetry and remote sensing. Especially data-driven approaches such as deep learning rely heavily on available annotated data. The amount of data is increasing significantly every day. However, reference data is not increasing at the same rate and finding relevant data for a specific domain is still difficult. Thus, it is necessary to make existing reference data more accessible to the scientific community as far as possible in order to make optimal use of it. In this paper we provide an overview of the development of our photogrammetry and remote sensing specific **Benchmark Metadata Database (BeMeDa)**. BeMeDa is based on MongoDB, a NoSQL database system. In addition, the development of a user-oriented metadata schema serves for data structuring. BeMeDa enables easy searching of benchmark datasets in the field of photogrammetry and remote sensing.

### 1. INTRODUCTION

The evaluation and comparison of newly developed methods and algorithms constitutes an important part of the scientific process. Benchmark datasets enable the development and testing of algorithms and thus make a valuable contribution to improve transparency and traceability (Long et al., 2020). Benchmark datasets and thereby benchmarking are used in numerous fields of applications, such as computer evaluation (Walters, 1976) or management applications (Išoraitė, 2004). Especially with the advent of deep learning, the importance of qualitative and quantitative data has increased (Munappy et al., 2019). Therefore the number of new extensive benchmark datasets constantly increases; this also applies to the remote sensing domain (Long et al., 2020).

#### 1.1 Photogrammetry and Remote Sensing domain

The popularity of benchmarking in photogrammetry and remote sensing is shown by the analysis of a literature database search by Bakula et al. (2019). For the evaluation of newly developed methods benchmark datasets are usually used. In the benchmarking process these datasets can be evaluated on the basis of their accuracy, sensitivity, effort and transferability by measuring, for example, execution time or memory requirements (Walters, 1976).

In order to perform such comparisons, benchmark datasets have to meet certain criteria compared to ordinary datasets (Hall, 2019). One of the key elements of benchmark datasets is the availability of reference data. For remote sensing, for example, this refers to the importance of the quality of annotated data (Long et al., 2020). However, compared to the available data, qualitative reference data are rare. Hence the availability of benchmark datasets is highly dependent on the application domain. For example, with the trend toward autonomous driving,

many of the benchmarks in photogrammetry and remote sensing address this environment, such as the well-known Kitti benchmark dataset (Geiger et al., 2012).

The challenge, however, is to find benchmark datasets relevant for one's own research, especially for young researchers (Brickley et al., 2019). In particular, very specific benchmark datasets exist in photogrammetry and remote sensing (e.g. multi-platform photogrammetry (Nex et al., 2015)) which are only known by a small expert group. Furthermore, there is a trend for such benchmark datasets to be of interest to researchers from other domains (Brickley et al., 2019). The current search options include, but are not limited to, querying scientific literature databases using corresponding keywords such as 'benchmark' and 'remote sensing' (Bakula et al., 2019; Long et al., 2020). However, this requires the description of a benchmark dataset in a published scientific paper with corresponding keywords that are subsequently added to the literature database (e.g. Scopus). In contrast, the Google Dataset Search offers the possibility to find open data by a web crawler, but relies on qualitative metadata (Brickley et al., 2019).

With these type of tools a specific effective benchmark dataset search in the photogrammetry and remote sensing domain is difficult. Furthermore, the search results have to be filtered manually. The use of general keywords such as 'remote sensing', 'photogrammetry' and 'benchmark' results in very different applications (Bakula et al., 2019). Moreover, not all found paper include open data and fit to the desired application. Additionally, metadata are not yet provided by all providers of specific photogrammetry and remote sensing benchmark datasets. Without such metadata, web crawler are unable to find the desired benchmark datasets.

Much more domain specific search tools focus on search options that are thematically relevant. For example, in the field of remote sensing, the EOD platform (Earth Observation Database, 2022)

\* Corresponding author

developed by IADF TC<sup>1</sup> allows filtering by sensors, tasks and locations. However, the included datasets are dependent on the input of the datasets by crowd working.

## 1.2 Related Work

For research and benchmarking suitable datasets are necessary. One possible source is to find them through search queries. Nevertheless, such a search is based on keywords and metadata (Chapman et al., 2020). This also applies to the Dataset Search from Google, developed since 2016 (Brickley et al., 2019; Benjelloun et al., 2020). Despite large growth, the datasets included depend on the quality and availability of metadata. With the increasing amount of data, the importance of using unique identifiers for datasets such as digital object identifier (DOI) is also increasing (Abe et al., 2014).

Databases of datasets can perform multiple tasks. One of them is to improve the usage of the datasets through a wider distribution or more extensive evaluations (Tohyama et al., 2008). Some of these databases are based on the acquired data by including them in the database (Quiring et al., 2016). Others, in contrast, are using only metadata in their database to improve the use and sharing of the data to avoid conflicts with ownership and permission (Abe et al., 2014; Lofstead et al., 2019; Tohyama et al., 2008). An example for a database related to the geospatial data domain is presented by Abe et al. (2014). To optimize databases for metadata, Lofstead et al. (2019) investigated NoSQL database systems.

For the collection and provision of datasets via metadata, the attributes they contain usually follow a predefined schema. For example, Brickley et al. (2019) uses, among others, the dataset type from schema.org (Data and Datasets - schema.org, 2021). Specialized databases use attributes developed for their purpose (Tohyama et al., 2008; Abe et al., 2014).

From the user's point of view, it is not enough just to create the databases and collect the information the datasets contain. Instead, the query functionalities have to be taken into account (Abe et al., 2014; Tohyama et al., 2008). This leads to different query possibilities and an appealing user interface (Tohyama et al., 2008).

## 1.3 Contribution

Our aim is to design and implement a database of benchmark datasets (**Benchmark Metadata Database, BeMeDa**) for remote sensing and photogrammetry applications. On the one hand, this should considerably simplify the search for suitable benchmarks, and on the other hand, increase the visibility of existing benchmarks in these research fields. In this regard, the use of benchmarks can be increased, thus improving comparability. A standardization of the properties is carried out by defining certain categories. This can be used to achieve a better comparability of different benchmarks and to facilitate the selection of suitable benchmarks. In particular, our database also considers datasets not already explicitly designated as benchmarks, but fulfill our criteria to be considered as such. The use of a NoSQL database allows flexibility to easily implement adaptations if necessary.

<sup>1</sup> Image Analysis and Data Fusion Technical Committee, <https://www.grss-ieee.org/technical-committees/image-analysis-and-data-fusion/>

This paper is organized as follows. At first we present an overview about database systems in section 2. In section 3 we introduce our procedure for the benchmark database. The implementation is presented in section 4. Finally, section 5 contains the discussion and section 6 the conclusion.

## 2. BACKGROUND

### 2.1 SQL versus NoSQL

Database systems are divided into relational and non-relational. While relational database systems make use of the SQL query language non-relational databases do not provide a SQL interface and are called 'Not only SQL' (NoSQL). (Meier and Kaufmann, 2019). Both database types store data. Thereby, entity-relationship (ER) models help to structure the data. For relational databases, such predefined schemas are necessary and fixed. Therefore, the model design must be done carefully. In contrast, NoSQL databases are schema-free (Meier and Kaufmann, 2019). Nevertheless, ER models represent the structure of data in an understandable and graphical way and thus also help to understand storage for NoSQL databases (Kaur and Rani, 2013). Furthermore, it is also possible to store unstructured data in NoSQL databases. The processing performance of NoSQL databases is more efficient, especially for a huge amount of data (Meier and Kaufmann, 2019). NoSQL database types are vertically and horizontally scalable. This allows to increase not only the number of elements entered in one server, but also to distribute them to several parallel servers (Kaur and Rani, 2013). In contrast to the previous advantages (e.g. high performance), a loss of consistency must be accepted with NoSQL (Meier and Kaufmann, 2019). Furthermore, there is no equivalent for NoSQL to SQL as a standard query language. Therefore, many users use APIs (Kaur and Rani, 2013).

### 2.2 Document stores

There are different specializations of NoSQL databases, such as key-value or column stores (Martins et al., 2021). Other types are document oriented stores. Thereby, documents save (semi-) structured data (Kaur and Rani, 2013; Meier and Kaufmann, 2019). Each document has its own identification value. In each document, the associated data is structured in a specific way e.g. using JSON format (Meier and Kaufmann, 2019). Thus, a very high flexibility is possible. The document ID is stored as a key-value pair and the data itself in a attribute-value manner (Meier and Kaufmann, 2019). Document databases are able to process many heterogeneous data (Meier and Kaufmann, 2019). Furthermore, only available attributes are included in the individual documents. If information is missing, empty fields can be omitted (Kaur and Rani, 2013).

There are numerous document oriented database software systems that can be used for free. These include, for example, Couchbase, CouchDB and MongoDB (Martins et al., 2021). The comparison presented by Martins et al. (2021) shows the best overall result for MongoDB software. Additionally, MongoDB is widely used. The graphical user interface 'Compass' support the use of MongoDB (MongoDB, 2021).

## 3. APPROACH

### 3.1 Benchmark criteria

As mentioned in section 1, it is necessary to distinguish between dataset and benchmark datasets. Accordingly, in this section the

benchmark criteria are defined. If a dataset meets these criteria, it can be included in our database. The starting point for the definitions is the contribution of Hall (2019). In our database a benchmark dataset:

1. includes reference data e.g. label, check points, etc.
2. has published results for a specified task e.g. classification accuracy
3. has a documentation of the data acquisition e.g. via published paper
4. provides free data access e.g. via dataset website

With this specifications it is possible to perform systematic evaluations and comparisons, for example with the use of public platforms (Long et al., 2020).

### 3.2 ISPRS keywords

Each ISPRS Congress paper identifies a few specified keywords. These keywords and corresponding papers were parsed exemplary for Commissions I to IV for the 2021 edition of the XXIV ISPRS Congress (ISPRS, 2021) (Figure 1) to determine important topics, the remote sensing and photogrammetry community is interested in. Due to the variety of different notations of the keywords the arrangement in groups is based on the count of the substrings of each keyword. In this regard the keywords and their corresponding scientific papers are extracted for each commission. As a result of the arrangement process keywords that exhibit the same count of substrings and a similar structure, i.e. upper vs. lower case as well as singular vs. plural form of the substrings belong to the same group. Based on the generated groups the corresponding count of scientific papers is summed up. Figure 1 visualizes the four most frequent keywords for each commission. The depicted keywords represent single or groups of keywords that contain different notations. However, this approach does not allow the creation of keyword groups if keywords exhibit a similar meaning but a different count of substrings. This causes the creation of unnecessary groups for keywords that exhibit a similar meaning (e.g. 'UAV' vs. 'Unmanned Aerial Vehicle (UAV)').

Based on this keyword grouping, an analysis of current and past trends may be performed. The frequency of the keywords 'machine learning' and especially 'deep learning' illustrates the commission-independent importance of reference and benchmark data.

### 3.3 Metadata

For the use of metadata depending databases, a predefined metadata schema simplifies the database construction. Hence, a suitable metadata scheme for photogrammetry and remote sensing has to be found. Possible evaluation options for sensors, geospatial data recorded by a specific sensor and algorithms are provided by Bakula et al. (2019). The keywords analysis from section 3.2 supports the statement by Bakula et al. (2019) and provides additional information, which metadata are relevant in the photogrammetry and remote sensing domain. In particular, the keyword analysis revealed that the use of different platforms (UAV, Sentinel-2 (Figure 1)) and machine learning tasks such as semantic segmentation are important. Various metadata are derived from this analysis, e.g. 'sensors' or 'platforms'. Relevant basic metadata attributes are derived from the 'Dataset' type

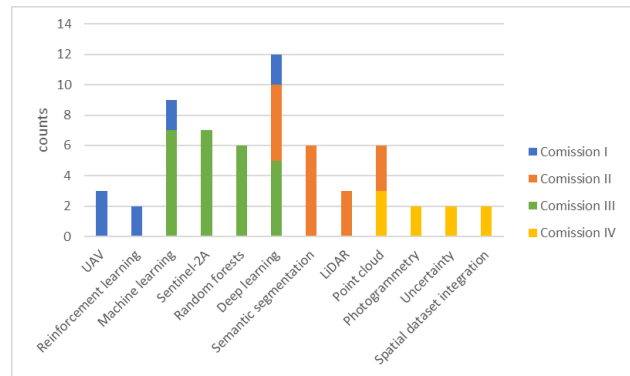


Figure 1. The four most frequent keywords extracted from papers in the annals of Commission I-IV of ISPRS Conference 2021. For the sake of clearness only one notation for the keywords is selected.

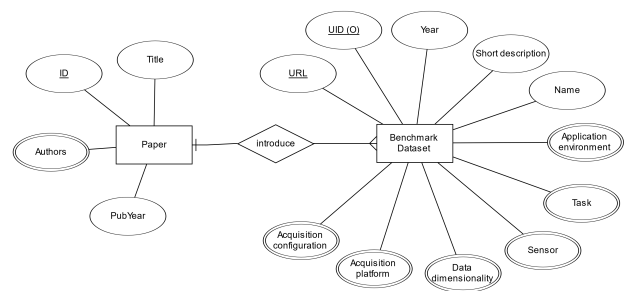


Figure 2. Entity-relationship model for metadata structure of benchmark datasets.

documented by schema.org (Data and Datasets - schema.org, 2021).

Based on this, our metadata considers the different requirements of the photogrammetry and remote sensing community. An ER model (Figure 2) was developed to assist with data structuring. This model illustrates that numerous of our attributes are multi-valued. In contrast to relational databases, document stores like MongoDB allow inserting such multiple values into a single attribute and no normalization is necessary.

In addition to Figure 2, Table 1 and Table 2 contain our metadata attributes. Thus the metadata distinguishes between different sensors and platforms. Furthermore, the dimensionality and special acquisition configurations specify the data properties. The areas of application for which a dataset is designed for are defined by the tasks. For further information, the paper presenting the dataset is included. More general attributes identify the benchmark dataset by its name, URL, publication year and, if available, an unique identifier (UID). The URL can also be used to provide a possibility to download the dataset. Finally, the environment and a short description provide an insight into the applications the benchmark dataset can be used for.

For a comparison to the existing schema.org attributes for datasets (Data and Datasets - schema.org, 2021), a brief presentation is given below. On the one hand schema.org metadata includes some general attributes, which are also included in our approach. For example, 'distribution' contains the URL link to the dataset download. The 'issn' however, does not reflect the common DOI as identifier. On the other hand the included dataset attributes only contain 'measurementTechnique' and 'variableMeasured',

Attribute	Description
Name	name of the benchmark dataset
Description	short description with details e.g. included classes
URL	link to the dataset information, partially download possible
UID	unique dataset identifier, e.g. DOI
Task	for which tasks the dataset is designed for
Sensor	which sensors are used for data acquisition
Acquisition Configuration	in which configuration the data are acquired
Acquisition Platform	on which platforms the sensors are mounted
Application Environment	in which environment the data was recorded
Dimension	which dimension has the provided data
Year	when was the benchmark published

Table 1. Defined relevant metadata attributes for benchmark datasets.

Attribute	Description
Title	title of the paper
ID	unique identifier, e.g. DOI
PubYear	year of publication
Authors	authors

Table 2. Defined relevant metadata attributes for additional paper, which introduces the benchmark and, in particular, evaluation results.

schema.org Dataset	our metadata
distribution	URL
issn	UID
measurementTechnique	Sensor
variableMeasured	Acquisition Configuration, Dimension

Table 3. Correspondences between schema.org dataset and our metadata.

Attribute	Attribute values
Acquisition configuration	multi-temporal multi-spectral multi-sensoral multi-view nadir oblique video
Sensors	structured light IMU GNSS RADAR LiDAR camera total station

Table 4. Possible attribute values for sensor and acquisition configuration attribute.

which are nonspecific and incomplete to represent all the different acquisition methods and measurements in photogrammetry and remote sensing. While the measurement technique can still be assigned to the sensor attribute, the measured variables are indirectly derived from the dimension, sensor and acquisition configuration attributes. The measured variables are often not explicitly stated and a certain level of expert knowledge is assumed. For our domain, a stronger differentiation of the properties facilitates the subsequent filtering in the search query (section 4.4). Table 3 contains an overview about the described correspondences between specific schema.org dataset attributes and our metadata schema. Moreover, schema.org dataset attributes are extended by further attributes inherited from parent object types e.g. 'abstract' as a type of short description. However, these attributes are also as basic as possible and do not reflect the desired search criteria.

The desired metadata information are often not directly available. Instead, the attribute values are extracted from the respective website or scientific paper. For our approach we introduce a generalization for the different multi-valued attributes. This simplifies the extraction process. To get an idea which attribute values the specified attributes accept, Table 4 provides some examples. The selected attribute values cover the common configurations in photogrammetry and remote sensing. Of course, adding any new values is possible.

## 4. IMPLEMENTATION

The implementation of the database consists of two main components: the Back-End and the Front-End. The Back-End contains the database itself and the Rest-API. The Front-End, on the other hand, contains a user-friendly interface for the communication with the Back-End. Below are the details about the respective implementations. A demo version of the implemented database will be linked on our website<sup>2</sup>.

### 4.1 Initializing the database

To initialize the database, a manual search for benchmark datasets was performed. We used a variety of keywords, like the ones presented in section 3.2, for manual search in established search machines to find datasets in our field. A short evaluation on the datasets was performed to check if they classify as benchmarks (section 3.1). Indications of this were, for example, the availability of training and reference data. The benchmark datasets were further analyzed to manually create metadata for each dataset according to section 3.3.

### 4.2 BeMeDa Back-End

The Back-End is primarily responsible for data management and processing. Our Back-End consists of two applications running simultaneously. The first one is the MongoDB database

<sup>2</sup> <https://www.geodesy.tu-darmstadt.de/fernerkundung/index.en.jsp>

```

    _id: ObjectId("61ae1f2883822a0d0b64e1d")
    title: "Hessigheim 3D - H3D"
    url: "https://ifpwww.ifp.uni-stuttgart.de/benchmark/hessigheim/default.aspx"
    year: 2021
    short_description: "11 classes, high density, high resolution, point cloud and textured me..."
    tasks: Array
  > sensors: Array
  > features: Array
  > 0: Object
  > feature: "acquisition_configs"
  > value: Array
  > 1: Object
  > feature: "area_application"
  > value: Array
  > 0: "urban environment"
  > 2: Object
  > feature: "acquisition_platform"
  > value: Array
  > 0: "uav"
  > 3: Object
  > feature: "dimensionality"
  > value: Array
  > 0: "3d"
  > 4: Object
  > feature: "ds_type"
  > value: Array
  > 0: "benchmark"
  > paper: Object
  
```

Figure 3. Example of a MongoDB benchmark document. The structure of the attributes in the 'features' field is based on the MongoDB attribute pattern.

and the second one is a python-based middleware application, which works as a link between the Front-End and the database. Furthermore we designed our Back-End regarding to REST-Architecture to ensure that our application is scalable for future extensions. In particular we used FastAPI-Framework to achieve this goal. After installation, new collections are defined. A collection summarizes multiple documents and is analogous to tables in relational databases (MongoDB, 2021). The first collection includes the benchmarks itself. The second collection contains the authors of the papers. To transfer the model from section 3.3 into MongoDB documents, the use of attribute pattern is particularly suitable. This primarily concerns on the similarities between multi-valued attributes. Exceptions in this case are the 'tasks' and 'sensors' categories. These two attributes are considered as mandatory. All other multi-valued attributes are subordinated in a common 'features' field. This means that fewer indexes are needed, which makes queries more efficient (MongoDB, 2021). An example of a benchmark document is presented in Figure 3. The collected benchmark datasets and related papers used to initialize the database are saved as CSV files. The processing of the data is done with Python using pymongo as an interface between MongoDB and Python. Thus, database queries are also made via this interface. Furthermore JSON objects are used as data exchange format between Back- and Front-End.

### 4.3 BeMeDa Front-End

For easy access to the database we built a Front-End with a graphical user interface. Therefore, we used HTML, CSS and Javascript and built our Front-End according to VueJS and Vuetify-Frameworks. With our graphic interface (Figure 4, Figure 5) it is possible to filter the database according to the attributes and attribute values, defined in section 3.3, via lists of checkboxes. The current state of our database is presented in Figure 4 that visualizes statistics about current parameters and their distribution of the already included benchmark datasets. Figure 5 demonstrates the results of a search query. In addition to the benchmark name, URL and publication year the accompanying information for tasks and sensors as well as a short description of the dataset and additional attributes are shown. A text-based input for the search for benchmarks is implemented as well. Currently, the text-based search is only possible with single words. The readout of the filter checkboxes is performed with a JavaScript function. This function saves the filter inputs as a JSON-file and hands it over to the Back-End. The JSON file contains the

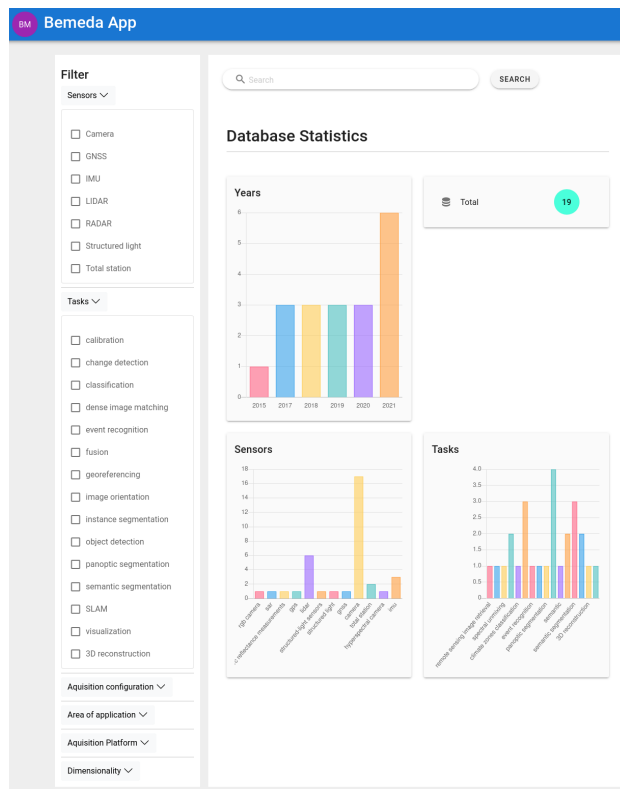


Figure 4. Example of our Front-End Application visualizing some statistics about the distribution of different parameters of - e.g. tasks, sensors, etc. of our benchmark database.

attributes and the attribute values set to true with the checkboxes. The Back-End processes the data and returns a JSON-file with the search result. The contents of this file are presented as a list of benchmark names with links to the complete lists of attributes.

### 4.4 Queries

As described in the previous section 4.3, there are two ways to search for benchmark datasets. The first possibility constitutes the use of the text search field. However, this search is currently limited to single inputs such as the name of the benchmark. The previous restrictions on free text search are based on the assumption that the user is unaware of the benchmark datasets. Thus, the main focus of the search is the selection of the desired filters, the second search option. Furthermore, it is possible to combine both search options. The different query cases are described below.

**4.4.1 Select none:** In this case, neither one of the filters is selected nor an entry is made in the search field. This keeps the search unrestricted. Thus the entire database is returned as the result of this query.

**4.4.2 Text search:** When using the text field search, the entire input is passed to the Back-End as one string. As already mentioned, to simplify further processing, the input is limited to only one search criterion. This allows searching for the entire string in the database. To find matches, correct spelling is necessary.

**4.4.3 Select filter:** In contrast to the previous cases, in this case the user selects some of the predefined filters. For example,

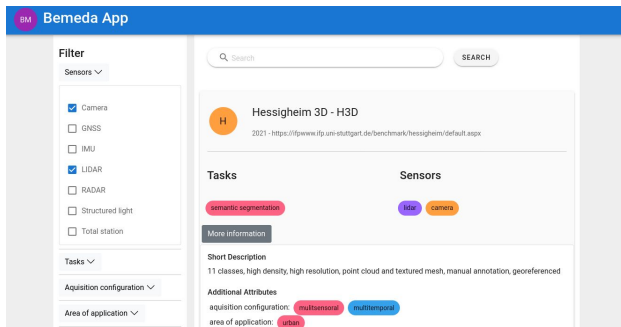


Figure 5. Example of a query with a search filter for all benchmarks with at least a Camera and a Lidar sensor.

it is desired that the benchmark data was captured with a camera from an UAV. With the filters selected accordingly, all datasets that meet these two criteria are filtered out. However, the filters do not have an exclusionary effect. With reference to the example, this means that data sets are also displayed that contain, for example, Lidar data in addition to the desired camera images.

**4.4.4 Select filter and text search:** Finally, by combining text search and filter selection it is possible to apply additional search criteria beyond the predefined filters. For example, a user selects multi-temporal data for indoor environment. By additionally entering e.g. a year of publication in the text field, the search results of the filter inputs can be limited to this year.

## 5. DISCUSSION

BeMeDa offers numerous advantages compared to the current more general search capabilities, because BeMeDa is optimized for the photogrammetry and remote sensing domain. The attributes defined in Table 1 are specialized for this area. This allows a much more targeted search than, for example, with the Google Dataset Search and is also more extensive than with the EOD Platform (Earth Observation Database, 2022). Our scheme also includes unique identifiers, in contrast to the EOD platform. This makes our database more robust in case a website link of a dataset changes. However, based on our defined attributes, some expertise is helpful. Nevertheless, the attribute value-based filters in the user-oriented Front-End make the search more accessible for the user, even to non-specialists.

BeMeDa includes not only the reference to the dataset itself, but also relevant metadata helping with selection of the benchmark dataset for specific application. For these metadata, there could be a need for frequent adjustment, which is related to the developments on the sensors market and progress of photogrammetric and remote sensing approaches. It requires high flexibility in updating the metadata structure. This requirement was considered while developing the structure of the database. By using NoSQL, the database can be adapted to future technologies and methods. Current development stage of BeMeDa, however, features some limitations and open tasks. Since the manual search for benchmark datasets and the compilation of the CSV file is very time consuming, BeMeDa so far only contains a small number of datasets. Thus, a high completeness cannot be achieved with this approach. This could be improved by the employment of advanced text processing tools including machine learning techniques. This issue, however, was not considered in the scope of this paper.

Further the development of BeMeDa has a large potential for the analysis of existing benchmarks and can support the photogrammetric and remote sensing community in defining gaps. On the one hand, these gaps may concern the publication process of benchmark datasets e.g. the use of metadata. On the other hand, as the size of the database increases, certain research fields can be identified where the selection of benchmark datasets remains difficult. These gaps can be used to create a road map for establishing new benchmark datasets.

## 6. CONCLUSION & FUTURE WORK

In this paper, we present an approach for collecting and searching benchmark datasets especially for photogrammetric and remote sensing applications. Previous search approaches for benchmark datasets are limited to high level of manual effort and, therefore, are usually very time-consuming. At the same time, the demand for data continues to grow strongly. In order to make progress in the development of new methods, it is essential to create comparable conditions. Therefore, the use of benchmark datasets is particularly important. With our BeMeDa presented here, we have succeeded in achieving a significant simplification in the search for benchmarks.

Nevertheless, there are numerous options to improve and extend BeMeDa. First of all, it is necessary, to make the search functionalities more flexible. Especially with an advanced processing of the text field input to enable the search for multiple attribute values as free text. The completeness of the database is of equal importance. In the future, we aim to add more benchmark datasets automatically or at least semi-automatically. The latter can be achieved, for example, by using a web form. This also allows to check the quality of the entries. Furthermore, it is possible to enrich the database with more information about the individual benchmark datasets. This includes, for example, details of the used methodology and an indication of the distribution in other papers. However, both are associated with an increased text processing effort.

## References

- Abe, S., Umemura, N., Koyama, Y., Tanaka, Y., Yagi, M., Yatagai, A., Shinbori, A., UeNo, S., Sato, Y., Kaneda, N., 2014. Progress of the IUGONET system - metadata database for upper atmosphere ground-based observation data. *Earth, Planets and Space*, 66(1), 133. <https://doi.org/10.1186/1880-5981-66-133>.
- Bakula, K., Mills, J. P., Remondino, F., 2019. A Review of Benchmarking in Photogrammetry and Remote Sensing. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-1/W2, 1–8. <https://doi.org/10.5194/isprs-archives-XLII-1-W2-1-2019>.
- Benjelloun, O., Chen, S., Noy, N., 2020. Google Dataset Search by the Numbers. J. Z. Pan, V. Tamma, C. d'Amato, K. Janowicz, B. Fu, A. Polleres, O. Seneviratne, L. Kagal (eds), *The Semantic Web – ISWC 2020*, Lecture Notes in Computer Science, 12507, Springer International Publishing, Cham, 667–682. [https://doi.org/10.1007/978-3-030-62466-8\\_41](https://doi.org/10.1007/978-3-030-62466-8_41).
- Brickley, D., Burgess, M., Noy, N., 2019. Google Dataset Search: Building a search engine for datasets in an open Web ecosystem. L. Liu, R. White (eds), *The World Wide*

- Web Conference*, ACM, New York, NY, USA, 1365–1375. <https://doi.org/10.1145/3308558.3313685>.
- Chapman, A., Simperl, E., Koesten, L., Konstantinidis, G., Ibáñez, L.-D., Kacprzak, E., Groth, P., 2020. Dataset search: a survey. *The VLDB Journal*, 29(1), 251–272. <https://doi.org/10.1007/s00778-019-00564-x>.
- Data and Datasets - schema.org, 2021. <https://schema.org/docs/data-and-datasets.html> (accessed 24.11.21).
- Earth Observation Database, 2022. <https://eod-grss-ieee.com/home> (accessed 16.3.22).
- Geiger, A., Lenz, P., Urtasun, R., 2012. Are we ready for autonomous driving? The KITTI vision benchmark suite. *2012 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, 3354–3361. <https://doi.org/10.1109/CVPR.2012.6248074>.
- Hall, M., 2019. What makes a good benchmark dataset? Agile Scientific . <https://agilescientific.com/blog/2019/4/3/what-makes-a-good-benchmark-dataset> (accessed 12.8.21).
- Išoraitė, M., 2004. Theoretical Aspects of Benchmarking Theory. *Public Policy and Administration*, 1, 21–26.
- ISPRS, 2021. <https://www.isprs.org/publications/annals.aspx/> (accessed 17.12.21).
- Kaur, K., Rani, R., 2013. Modeling and querying data in nosql databases. *2013 IEEE International Conference on Big Data*, 1–7. <https://doi.org/10.1109/BigData.2013.6691765>.
- Lofstead, J., Ryan, A., Lawson, M., 2019. Adventures in NoSQL for Metadata Management. G. Juckeland (ed.), *High performance computing*, Lecture Notes in Computer Science, 11887, Springer International Publishing, Cham, 227–239. [https://doi.org/10.1007/978-3-030-34356-9\\_19](https://doi.org/10.1007/978-3-030-34356-9_19).
- Long, Y., Xia, G., Li, S., Yang, W., Yang, M. Y., Zhu, X. X., Zhang, L., Li, D., 2020. DiRS: On Creating Benchmark Datasets for Remote Sensing Image Interpretation. *CoRR*, abs/2006.12485. <https://arxiv.org/abs/2006.12485>.
- Martins, P., Morgado, F., Wanzeller, C., Sá, F., Abbasi, M., 2021. MongoDB, Couchbase, and CouchDB: A Comparison. Á. Rocha, H. Adeli, G. Dzemyda, F. Moreira, A. M. Ramalho Correia (eds), *Trends and Applications in Information Systems and Technologies*, Advances in Intelligent Systems and Computing, 1366, Springer International Publishing, Cham, 469–480. [https://doi.org/10.1007/978-3-030-72651-5\\_45](https://doi.org/10.1007/978-3-030-72651-5_45).
- Meier, A., Kaufmann, M., 2019. *NoSQL Databases*. Springer Fachmedien Wiesbaden, Wiesbaden, 201–218. [https://doi.org/10.1007/978-3-658-24549-8\\_7](https://doi.org/10.1007/978-3-658-24549-8_7).
- MongoDB, 2021. <https://www.mongodb.com/> (accessed 9.12.21).
- Munappy, A., Bosch, J., Olsson, H. H., Arpteg, A., Brinne, B., 2019. Data management challenges for deep learning. *2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, 140–147. <https://doi.org/10.1109/SEAA.2019.00030>.
- Nex, F., Gerke, M., Remondino, F., Przybilla, H.-J., Bäumker, M., Zurhorst, A., 2015. ISPRS Benchmark for Multi-Platform Photogrammetry. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3/W4, 135–142. <https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/II-3-W4/135/2015/>.
- Quiring, S. M., Ford, T. W., Wang, J. K., Khong, A., Harris, E., Lindgren, T., Goldberg, D. W., Li, Z., 2016. The North American Soil Moisture Database: Development and Applications. *Bulletin of the American Meteorological Society*, 97(8), 1441–1459. <https://doi.org/10.1175/BAMS-D-13-00263.1>.
- Tohyama, H., Kozawa, S., Uchimoto, K., Matsubara, S., Isahara, H., 2008. Construction of a Metadata Database for Efficient Development and Use of Language Resources. *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, European Language Resources Association (ELRA), Marrakech, Morocco. [http://www.lrec-conf.org/proceedings/lrec2008/pdf/428\\_paper.pdf](http://www.lrec-conf.org/proceedings/lrec2008/pdf/428_paper.pdf).
- Walters, R. E., 1976. Benchmark techniques: a constructive approach. *The Computer Journal*, 19(1), 50–55. <https://doi.org/10.1093/comjnl/19.1.50>.