DEVELOPMENT OF A MULTI-DIMENSIONAL ARRAY DATABASE BASED MASSIVE SATELLITE INFORMATION PROCESSING AND ANALYSIS SYSTEM: KIWI-SAT

S. Han[1] *, J. Han[1], Y. Choi[1], S. Lee[1], M. Kim[1], S. Lee[1]

[1]KARI, 169-84 Gwahak-ro, Yuseong-gu, Daejeon, Korea – (shan, jhan, choiyj, jelline15, mukim, leesg)@kari.re.kr

Commission IV, WG IV/4

KEYWORDS: KIWI-Sat, Big data, KOMPSAT-3/5, SAR, AI analysis, Array Database, Rasdaman.

ABSTRACT:

Information and communication technology (ICT) is mainly applied to finance, telecommunications, and public sectors. However, since the early 2010s, there have been efforts to apply ICT to various fields such as aerospace, life science, energy, and automobiles. Recently, artificial intelligence and big data technologies have also been applied in the aerospace field, among others. In the field of aerospace, earth observation attracts the most interest.
One reason earth observation attracts such interest is that the availability of a satellite constellation allows more frequent observations of objects of interest. Another reason is the improvement of technologies that can process massive satellite images such as data cubes for urban change detection, disaster monitoring, and traffic analysis. When performing earth observation using satellite images, it is necessary to process a large amount of satellite information in real-time and analyze satellite images with artificial intelligence. Such research is continuing in various ways. The field of interest in this study is the processing technology for storing and retrieving large volumes of satellite images. Rasdaman and SciDB were two available open-source software applications.
In this paper, KIWI-Sat, which processes and analyzes a large amount of satellite imagery, mainly described and also show two KIWI-Sat demos with AI algorithms that are developed for Korean satellite images in KARI, one is super resolution algorithm of K3 and the other is water segmentation algorithm of K5, SAR satellite image.

1. INTRODUCTION

Information and communication technology (ICT) is mainly applied to finance, telecommunications, and public sectors. However, since the early 2010s, there have been efforts to apply ICT to various fields such as aerospace, life science, energy, and automobiles.

Recently, artificial intelligence and big data technologies have also been applied in the aerospace field, among others. In the field of aerospace, earth observation attracts the most interest. One reason is that the availability of a satellite constellation allows more frequent observations of objects of interest. Another reason is the improvement of technologies that can process massive satellite images such as data cubes for urban change detection, disaster monitoring, and traffic analysis.

When performing earth observation using satellite images, it is necessary to process a large amount of satellite information in real-time and analyze satellite images with artificial intelligence. Such research is continuing in various ways. The field of interest in this study is the processing technology for storing and retrieving large volumes of satellite images. Rasdaman and SciDB were two available open-source software applications. However, Paradigm4, the developer of SciDB, does not provide it as open-source software anymore.

Since 2018, based on multi-dimensional array database technology, we have been developing KIWI-Sat, a system that can store, process, and analyze the images of Korean satellites such as KOMPSAT-2(K2), KOMPSAT-3(K3), KOMPSAT-5(K5).

In this paper, we describe KIWI-Sat in section 3 and show two KIWI-Sat demos in section 4. Both were done with AI algorithms that were developed in KARI (Korea Aerospace Research Institute) for Korean satellite images. One is the super-resolution algorithm of K3 (Choi et al., 2021) and the other is the water segmentation algorithm of K5, a SAR (Synthetic Aperture Radar) satellite image (Kim et al., 2021).

2. RELATED WORKS

In earth observation with satellite images, the technology for storing and processing massive satellite images is mainly divided between a Hadoop-based system (Boudriki Semlali, 2021, Nguyen, 2021) and a multidimensional array database system. See Figure 1 (Baumann, 2014, Rodriges Zalipynis, 2017, Baumann et al., 2021).

There are also systems that process satellite images on demand as opposed to storing them in a database or with Hadoop (Appel, 2019).

The multidimensional array database system and the Hadoop-based systems are equally effective in searching some areas of satellite images, but the database system has the advantage of easy management and fast development. However, in both systems the techniques for converting from NetCDF and HDF file formats used in satellite images to their own formats is very time-consuming. This must be considered when developing a big data system for earth observation.

Since early 2019, there are no public sites for downloading the SciDB community version. Paradigm4 (Paradigm4, 2022) does not support SciDB as open-source software anymore.

* Corresponding Author

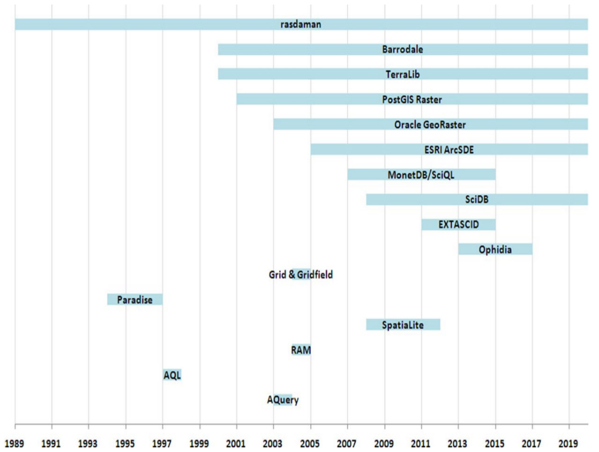

Figure 1. Early history of array database systems (Baumann, 2021).

In Figure 2, Paradigm4 shows REVEAL, which is an Agile Science Platform Designed for Scientific Data and Scientific Computing based on SciDB.

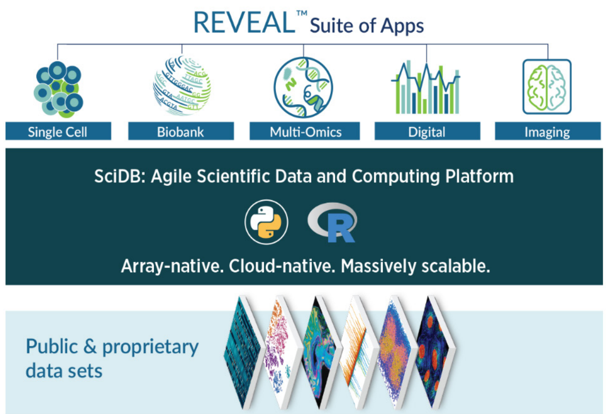

Figure 2. REVEAL & SciDB in Paradigm4.

In section 3.3, we describe simple comparative performance tests of uploading and searching satellite images between Rasdaman and the SciDB array database system.

3. KIWI-SAT

Since 2018, we have been working on developing KIWI-Sat, a system for processing and analyzing massive satellite information especially supporting Korean satellite images such as K2, K3, K5. KIWI-Sat mainly uses Python as a development language in the Ubuntu environment, and uses a number of open-source software applications such as Rasdaman which is a multi-dimensional array database system, Pytorch which is an AI developing framework, Django which is a web developing framework, and Mapbox which is a map service software KIWI-Sat supports GeoTiff, HDF 5, and JP2 which are the main data types of representative satellite images.

KIWI-Sat is mainly being developed to support Korean satellite images such as KOMPSAT-2, KOMPSAT-3, KOMPSAT-3A, KOMPSAT-5, and GOCI as well as, overseas satellite images such as Sentinel 1A, Sentinel 1B, Sentinel 2B, SPOT, and PlanetScope. Because KIWI-Sat supports various satellite images, other satellite images can easily be supported.

In this section, we describe the system architecture of KIWI-Sat, its main features, and brief comparative test results between Rasdaman and SciDB.

3.1 System Architecture of KIWI-Sat

Figure 3 shows the deployment diagram of KIWI-Sat. KIWI-Sat is being developed using a number of open-source software such as Rasdaman, Pytorch, Django, and Mapbox.

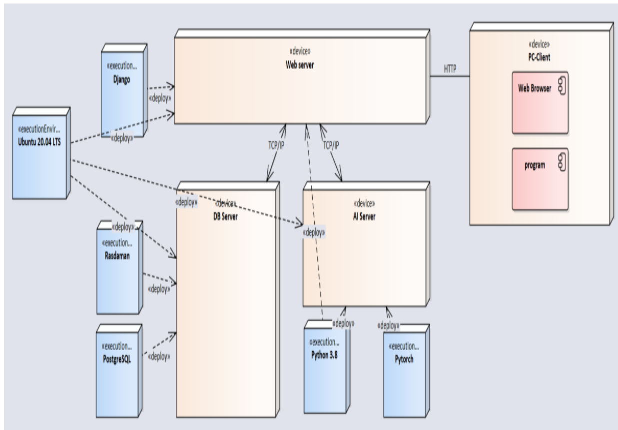

Figure 3. Deployment Diagram of KIWI-Sat.

It uses Rasdaman to process massive Raster-based satellite information, Pytorch to process AI inference module, and Django and Mapbox for visualizing and overlaying Satellite images. KIWI-Sat is mainly composed of five subsystems.

K-SDA (KIWI-Sat Data Access): process satellite information based on Rasdaman, an array database system

K-SAA (KIWI-Sat AI Analysis): analyze AI-based satellite images

K-SVI (KIWI-Sat Visualization): visualize satellite image based on map service

K-SUT (KIWI-Sat Utility): provide utility functions such as system resource monitoring, and upload and download of original satellite images

K-SOA (KIWI-Sat OpenAPI): provide OpenAPI for satellite information access

The main components of KIWI-Sat are K-SDA and K-SAA.

K-SDA was developed using Rasdaman. In processing satellite images, the multidimensional array database system can easily handle storing and searching of arrays with the SQL standard, and supports the processing of massive raster data over Petabytes.

For supporting many satellite images such as the KOMPSAT series and other satellite images in KIWI-Sat, we need to have satellite-related meta-information. Table 1, the first of 8 tables, is the data model of KIWI-Sat. We briefly describe the database tables.

The main tables are raster_band_info which has band information of stored satellite images and raster_set_metadata which stores metadata for satellite images.

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLVIII-4-W1-2022-181-2022 | © Author(s) 2022. CC BY 4.0 License.

Table	Description
data_expiration_info	information for managing expiration time for each satellite image device
detected_object_info	information for detected object with AI tools (for future use)
raster_band_info	band information of each satellite image data
raster_set_metadata	metadata for satellite image files
overlay_files_access	metadata for managing upload history information of satellite image overlay images
overlay_files_metadata	metadata of overlay images for multi-zoom level for each satellite image device
scheduler_task_log	metadata for managing the status of satellite image input/output automation tasks
task_status_info	information for managing the status of satellite image input/output automation tasks

Table 1. Tables of KIWI-Sat.

K-SAA is linked to the AI module for analyzing the satellite image. KIWI-Sat has a structure that can be easily linked with the inference code and the parameters of the artificial intelligence module. It works with the K-SVI by receiving the satellite image obtained from K-SDA and transmitting the obtained result to the Django web framework by executing the AI inference code.

3.2 KIWI-Sat Main features

3.2.1 Upload and Download Satellite: KIWI-Sat provides the upload and download function of satellites. There are two steps in uploading satellite images; the first is done on the console which is connected to KIWI-Sat by a ssh (secure shell) terminal and the second is done on the menu of a web browser such as Chrome.

3.2.2 Map based Functions: KIWI-Sat provides various web-based functions on map such as 'move to target location' using keywords or pair of longitude and latitude, 'ROI (region of interest) search' with any size of rectangle, and 'choose satellites' for selecting certain satellites. Figure 4. shows an example of KIWI-Sat.

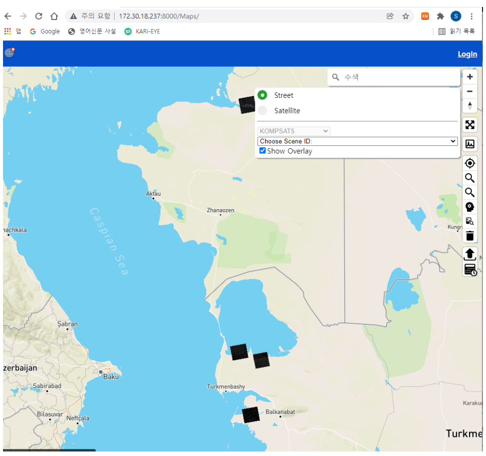

Figure 4. Screen Example of KIWI-Sat.

3.2.3 Overlay Satellite Image: For handling real data of target satellites by array, we need a simulated reference satellite image, just for visualization. For this, KIWI-Sat provides overlay functions. When satellite images are uploaded into KIWI-Sat, it makes overlay files of each satellite and manages them. Table 2. shows the zoom level of each satellite.

Satellite	Low	High
K2	9	16
K3	9	16
K3A	9	17
Sentinel-1B	9	12
SPOT	11	15

Table 2. Zoom level of target satellite for overlay.

3.2.4 REST API: KIWI-Sat provides some REST API for handling and developing program. Table 3. shows a list of KIWI-Sat REST API.

Name	Description
clip_image	get clip image
roi_image	get roi image
roi_tif_image	get roi tiff image
sat_metadata	get metadata of satellite image
satellite	get information of satellite image such as total count, satellite name etc.
tilemap	get tilemap image
tilemap_history	get tilemap history

Table 3. REST API of KIWI-Sat.

REST API can be executed in a browser and some code. Figure 5. shows one example of clip_image REST API.

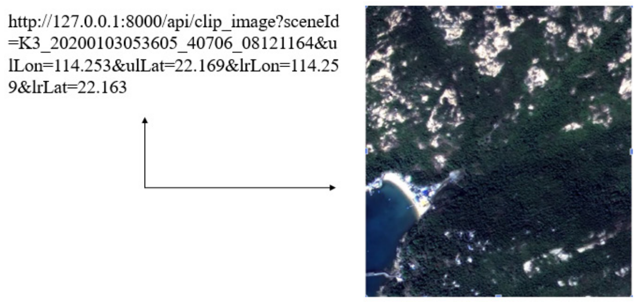

Figure 5. Example of clip_image REST API.

3.3 Rasdaman vs. SciDB performance comparison

At the beginning of KIWI-Sat development, SciDB was used as the multi-dimensional array database system, but as SciDB ended its open-source policy in early 2019, we needed to it with another system.

We investigated a few array database systems and selected Rasdaman as a candidate system. Before applying Rasdaman to KIWI-Sat, we checked its functions, and tested its performance.

We used Rasdaman 10.0 and SciDB 19.11 for the test. The test environment is given in Table 4.

Type	CPU	RAM(GB)	OS
Host	Intel(R) Xeon(R) CPU E5-2637 v3 @ 3.5 GHz	128	Windows Server 2012 R2 Standard

VM 1	8 core	32	Ubuntu 20.04
VM 2	8 core	32	CentOS 7.5

Table 4. Test Environment.

Table 5. and Figure 6. show test data information and ROI (Region of Interest) area with size of 1200x1000 pixels. The test satellite image is K3. The file size for upload is 1.56 GB and the band count is 1.

Item	Content
Satellite Type	K3
File Size	1.56 GB
Band Count	1
ROI Size	1200 x 1000

Table 5. Test Satellite Image.

Figure 6. ROI Area for Search performance Test.

Table 6. shows the Test results. Rasdaman is better than SciDB in uploading the satellite images and storing them in the Database. The search performance for the selected area was better in SciDB. Considering the real-time performance of recent satellite image analysis, it was confirmed that Rasdaman performed better.

Test	Rasdaman	SciDB
T1. upload(sec)	37.9	1,839
T2. Search(sec)	1.96	0.99

Table 6. Test Result of Upload and Search

4. APPLICATION EXAMPLE

We installed the KIWI-Sat at the HP server on KARI. The KIWI-Sat was installed in Ubuntu 20.04 LTS, and it stores K2, K3, K3A, K5 satellite images.

In this section, we show the execution result of two AI algorithms that were developed in KARI using KOMPSAT satellite images in the KIWI-Sat. One is the super-resolution algorithm of K3 (Choi et al., 2021) and the other is the water segmentation algorithm of K5, SAR satellite image (Kim et al., 2021).

Table 7. shows the test satellite image specification for KIWI-Sat test with 2 AI algorithms.

	K3	K5
Satellite Type	Optics	SAR
Spatial Resolution	PAN : 0.7 m MS: 2.8 m	2.5x2.5 m (Range/Azimuth)
Image Count	8	10
Area	Hong Kong	Water area, not specific region

Table 7. Satellite Image Specification for KIWI-Sat Test.

4.1 KOMPSAT-3 Satellite Image – Small Vehicle Detection Performance with Super Resolution

Figure 7. shows the architecture of super-resolution network for K3. For details, refer to Choi et al, 2021. We integrated this algorithm into KIWI-Sat.

To compare the performance improvement according to the super resolution technique, we analyzed the small vehicle detection performance difference depending on whether super resolution method is applied to K3 image.

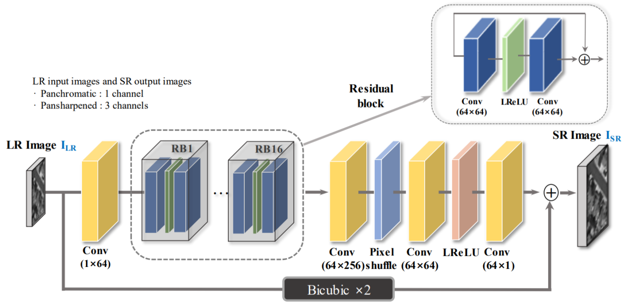

Figure 7. Architecture of K3 Super Resolution Network.

This result confirms that the super-resolution technology can improve the small object detection performance.

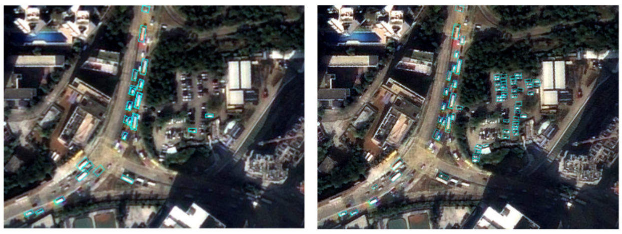

Figure 8. (left) object detection result on original KOMPSAT3, (right) object detection result on super resolution image

4.2 KOMPSTAT-5 Satellite Image - SAR Water Segmentation

We used a SAR water segmentation algorithm, which was developed in KARI for K5 satellite image; the High Resolution Network (HRNet) model (Wang et al, 2020) shown in Figure 9 was used along with other models. For details, you can refer to (Kim et al, 2021, NIA 2022).

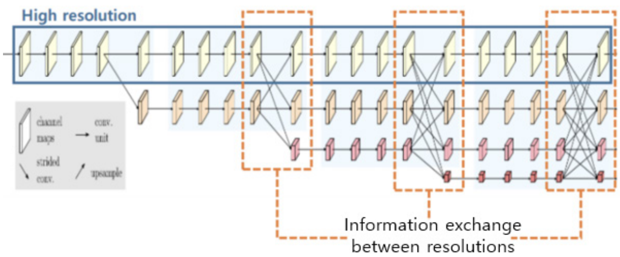

Figure 9. HRNet model

Figure 10. shows the area of K5 SAR images for the water segmentation test. We selected various test satellite images such as mountain water areas, agricultural water areas, complicated water areas, and urban water areas.

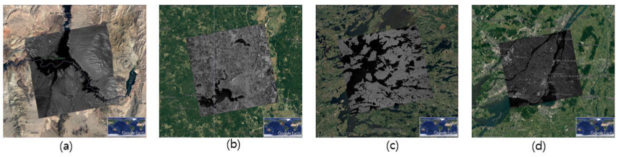

Figure 10. (a) mountain water area, (b) agricultural water area, (c) complicated water area, (d) urban water area

Figure 10. shows the result of the ROI search. The process is as follows. First, choose a K5 satellite image, move to the target location, and select a rectangular area with the mouse. Then click the ROI search button, and KIWI-Sat returns the original data that was delivered from Rasdaman.

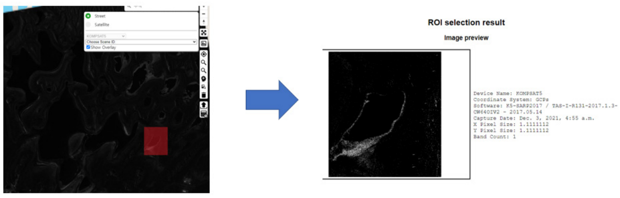

Figure 11. ROI select page

Table 8. shows the result page of water segmentation in the K5 image.

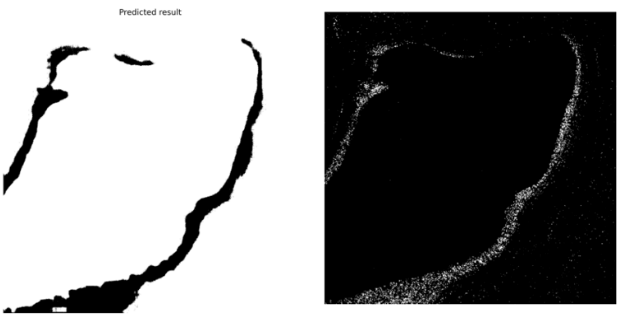

Table 8. The result of SAR Water Segmentation

5. CONCLUSIONS

We have looked at KIWI-Sat, massive satellite image processing and analysis system, and the demo that was applied two AI modules.

First, in KIWI-Sat, we described the system architecture of KIWI-Sat, its main features such as the upload-download utility, map-based ROI, search location with keywords and we described performance test results between Rasdaman and SciDB. Then we described the REST API list and explained through examples how to develop some codes.

Second, we showed two KIWI-Sat demos with AI algorithms that were developed for Korean Satellites in KARI. One is the super-resolution algorithm of K3 and the other is the water segmentation algorithm of K5, which is a SAR satellite image.

We hope that this technology will be used in earth observation. Further development and enhancement of KIWI-Sat is however needed. Although KIWI-Sat is not currently under construction, we plan to develop technologies for searching between multiple satellite images and extending satellite images to multiple nodes. We will construct a testbed for a private cloud based on OpenStack.

ACKNOWLEDGEMENTS

This study was supported and funded by the Satellite Data Applications project of the Ministry of Science and ICT (No. 1711170906). I thank Realtimetech and Kim, J., the team manager, and the team members of Realtimetech for their expertise and assistance throughout our study.

REFERENCES

Appel, M., and Pebesma, E., 2019: On-Demand Processing of Data Cubes from Satellite Image Collections with the gdalcubes Library in *Data* 4, no. 3: 92, doi.org/10.3390/data4030092

Baumann, P., 2014: rasdaman: Array Databases Boost Spatio-Temporal Analytics in *2014 Fifth International Conference on Computing for Geospatial Research and Application*, 54-54, doi.org/10.1109/COM.Geo.2014.1.

Baumann, P., Misev, D., Merticariu, V. et al., 2021: Array databases: concepts, standards, implementations in Journal of Big Data 8, article number: 28, doi.org/10.1186/s40537-020-00399-2

Boudriki Semlali, B.-E., and Freitag, F., 2021: SAT-Hadoop-Processor: A Distributed Remote Sensing Big Data Processing Software for Earth Observation Applications in *Applied Sciences* 11, no. 22: 10610, doi.org/10.3390/app112210610.

Choi, Y., Han, S., Kim, Y., 2021: A No-Reference CNN-Based Super-Resolution Method for KOMPSAT-3 Using Adaptive Image Quality Modification in *Remote Sens.* 2021, *13*, 3301. doi.org/10.3390/rs13163301

Esteves, A., and Pina, A., 2012: A WCS-BASED APPROACH TO INTEGRATE SATELLITE IMAGERY DATA IN WILDFIRE SIMULATION in Proceedings of the 8th International Conference on Web Information Systems and Technologies - WEBIST, 238-241, doi.org/ 10.5220/0003898002380241

Kim, M.U., Oh H., Lee, S.J., Choi, Y. and Han, S., 2021: A Large-Scale Dataset for Water Segmentation of SAR Satellite in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 9796-9801, doi.org/ 10.1109/IROS51168.2021.9635863.

Nguyen, D., and Le, H.A., 2020: A Big Data Framework for Satellite Images Processing using Apache Hadoop and RasterFrames: A Case Study of Surface Water Extraction in Phu Tho, Viet Nam in International Journal of Advanced Computer Science and Applications (IJACSA), 11(12), doi.org/10.14569/IJACSA.2020.0111289.

NIA, 2022: Apply Deep Learning Based Water Detection Algorithm, https://niasatellitedata.imweb.me/service_sar (11 June 2022)

Paradigm4, 2022: Platform overview, https://www.paradigm4.com/technology/scidb-platform-overview/ (11 June 2022).

Rasdaman, 2022: Rasdaman 10.1 Documentation, https://doc.rasdaman.org/01_introduction.html (7 June 2022).
Rodriges Zalipynis, R.A., Pozdeev, E., Bryukhov, A., 2017: Array DBMS and Satellite Imagery: Towards Big Raster Data in the Cloud. In Analysis of Images, Social Networks and

Texts(AIST 2017), 269-279, doi.org/10.1007/978-3-319-73013-4_25

Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., Liu, W., Xiao, B., 2020: Deep High-Resolution Representation Learning for Visual Recognition in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 10, 3349-3364, doi.org/10.1109/TPAMI.2020.2983686.