DRONE-BASED CONTAINER CRANE INSPECTION: CONCEPT, CHALLENGES AND PRELIMINARY RESULTS

Mehdi Maboudi^{*}, Ahmed Alamouri, Vanessa De Arriba López, Mohammad Shafi Bajauri, Cosima Berger, Markus Gerke

Institute of Geodesy and Photogrammetry, Technische Universität Braunschweig, (m.maboudi, m.gerke)@tu-braunschweig.de

Commission II, WG II/10

KEY WORDS: Crane Inspection, Structural Health Monitoring, Drone, Defects, Database, Machine Learning

ABSTRACT:

Container crane inspection is a very important task to maintain their uninterrupted operation. Nevertheless, this is a costly and timeconsuming activity if performed manually. Recently, image-based detection of surface damages or changes using drones has gained increasing interest in industry; especially when objects of interest have a complex structure like container cranes. One main aim of this paper is a single-epoch image analysis which will also serve later for multi-epoch processing. It provides reliable information about current defects that may lead to big damages if not inspected by experts. Naïve Bayes classifier is employed to classify the images in different classes of which critical defects and especially rust is important. The preliminary results show that the precision on the target class reached about 99%. However, 87% percent recall in this class is not enough and it should be improved for this application. Having a large dataset requires an efficient data management system to provide users and decision makers with the information needed. In addition, in order to foster full automation, the aforementioned image analysis component should have a direct connection to the database and thus is able to query image and semantic information. We therefore introduce the second aim of our research, that is a concept for database design. Here, not only the raw data and the final results are integrated but also the intermediate results. At the same time, the database concept is connected to an integrated client interface that allows retrieving data of interest in a virtual globe.

1. INTRODUCTION

Health monitoring is an essential process in ensuring the safety and serviceability of civil infrastructure like bridges and container cranes (Rao et al., 2020; Saleem et al., 2020; Stein, 2018). Current practice for assessing structural health of container cranes is mainly based on visual inspections by human operators (Hoskere et al., 2020). Container bridges or ship-toshore cranes are the common means in seaport container terminals for loading and unloading the containers from container ships. To maintain the uninterrupted and all-day operation of container cranes in a seaport, it is important to carry out a thorough and reliable inspection.

Inspections are time-consuming; there are numerous places that can only be reached physically and manually with great efforts. Changes in the surface of the container cranes (e.g. colour irritations, surface bulges, rust accumulation) must be detected at an early stage, as massive consequential damage or even breakage of the container cranes can occur. The breakage of the container crane at the NTB container terminal in Bremerhaven on May 15, 2015 during loading of "Maersk Karachi" is a dramatic example of the consequences that an unrecognised weak point in the structure can have.

In most seaports, industrial climbers are used, among other means, to inspect certain parts of the container crane in detail. Due to the associated safety requirements and the port-specific circumstances (strong winds, frost periods, strongly changing daylight incidence), the work is costly and risky. The laborious, time-consuming, unsafe and subjective nature of manual inspections motivate research into methods for automating such inspections. Drones have become an efficient tool in many applications due to the flexible use in data capturing process (Kerle et al., 2019; Nex & Remondino, 2014; Sahebdivani et al., 2020). Therefore, a logical step forward in increasing the automation of container cranes' inspection is employing the photos taken by drones and the visual evaluation of the photos with the help of qualified specialists. However, this approach is still subjective and based on personal assessment, experience and the respective daily form of the operators. In addition, the volume of data will become larger over time so that a manual investigation will become increasingly difficult to perform in the terms of time, cost and capacity.

Another very important aspect of health monitoring of the container cranes is the temporal analysis of the captured images. Automatic comparison of changes in the same areas and regions of the container cranes over a longer period of time could be done more efficiently using automated intelligent image understanding approaches. If the images, evaluation results and annotations would be stored in a powerful database, documentation of suspected defects would be improved as well as maintenance processes, including provision of security with regard to liability issues.

Based on the above-mentioned thoughts and requirements, during a joint collaboration between Hamburger Hafen und Logistitik AG (HHLA - a large operator of the seaport in Hamburg, Germany) and the Institute for Geodesy and Photogrammetry at the Technical University of Braunschweig, a research project called "*ABC-Inspekt*" has been initiated with the aim of automated and intelligent analysis of drone-based images to facilitate the inspection of container cranes. In this paper, the concept, challenges and preliminary results of the project are presented.

^{*} Corresponding author

2. CONCEPT AND METHODS

2.1 Overall concept

In order to realise visual inspection based on drone images, several side constraints need to be considered. Flight planning and capturing of images for complex container bridges needs to consider many parameters. The areas which are prone to defects are called "neuralgic areas" in the following. Only those specific parts of the crane need to be monitored regularly, such as junctions and bolting areas. Examples of those neuralgic areas are shown later. Those areas embody quite complex 3Dstructures with varying orientation of planes within the surface. In terms of flight planning this means that a constant image resolution at the object (GSD: ground sampling distance) is very difficult to obtain. In addition, the trade-off between GSD and footprint size needs to be regarded, also keeping in mind that a capturing of overlapping images is difficult to obtain fully automatically close to those massive steel structures (GNSS outages, multipath, magnetic field disturbances). Currently, experienced pilots are flying manually, and the further automation of navigation will be subject to upcoming research.

The research goals are formulated in a way to address the needs of the operators at a modern seaport who have the task to organise the maintenance of container bridges. Figure 1 depicts important components of the overall concept.



Figure 1. Overall concept of the project

As an example, some GSD values are computed based on different distances to a neuralgic area with respect to three standard lenses of the camera Sony Alpha 7R IV (Table 1).

Focal length	Distance to	GSD	Footprint
(mm)	object (m)	(mm/pix)	$(m \times m)$
85	15	0.66	6×4
	25	1.10	11×7
50	15	1.13	11×7
	25	1.88	18×12
35	15	1.60	15×10
	25	2.68	26×17

Table 1. Parameters based on 3 standard lenses of the camera

 Sony Alpha 7R IV with different distances to object

Another pitfall is that within those distances the focus of the camera is not driven to infinity, that is, the individual depth of field is varying per view. As a result, focussing on the area of interest means changing the effective focal length. Since the main aim is to derive sharp images and to analyse semantic image content rather than very accurate geometries, we do not insist on

using a constant focus. Ultimately, after the flight, a set of images for a certain container bridge is captured and clustered according to the named neuralgic areas.

An important requirement concerning the database concept is that the images and results from automatic image analysis are accessible to the human operator in order to allow manual intervention in a user-friendly environment, i.e. a modern GUI (graphical user interface). One pillar of the image processing steps is that a dual strategy is pursued: on the one hand, images from each single epoch are analysed in order to find hints for possible defects, and on the other hand the comparison of current images with historic images (multi-epoch) is assumed to add reliability to the entire process. In this paper, however, we are focussing on the single epoch case only.

2.2 Image analysis

Pixel-based and object-based image analysis are two possible solutions for very high-resolution image classification (Blaschke et al., 2014). Considering objects as the base processing units and utilising shape features and inter-objects relations can help in separating the segments in feature space. Moreover, image objects can be related to real objects, more closely. However, the quality of initial image segmentation has a significant influence on the results (Maboudi et al., 2017). Hence, at the current step of this research pixel-based approaches are utilised.

Using any of above-mentioned strategies, the results could not be error free. Some possible reasons could be inter-class similarity and intra-class heterogeneity of the dataset, uncontrollable environmental parameters and possible inconsistencies between model assumptions and real mapping between input and output space. Therefore, it is crucial that we consider the type of errors in evaluation of the classifiers, especially for a target class (defect). Although in ideal case, we would like to have very high values for both recall and precision measures; in practice this it is not possible due to bias-variance trade-off. In this project FN (false negatives; overlooked suspicious points) are more important than FP (false positives; falsely indicated defects). While FP ultimately leads to higher cost and reduced efficiency, FN is very critical and results in lower reliability of the approach which could have very dangerous consequences. The workflow of the image processing module for defect detection is represented in Figure 2, where the main objective is automatic detection of cranes' damages or signs that show that the crane part should be inspected for possible defect before future damages occur.

Once images are ready for the processing, the first step is to create a training dataset. The generated training set is used to train and evaluate the classifier for detecting the defect.

Parallel to training the classifier, a foreground-background separations step is applied to the images to decrease the complexity of the classification problem. After training the classifier and separating foreground from background, classification can be applied on the unseen images and pseudocoloured images are obtained according to the predefined classes. Afterwards, a visual inspection is employed in order to evaluate the performance of the classifier on unseen datasets.



Figure 2. Workflow of the image processing module

2.2.1 Foreground separation: The main goal of this step is to split the image into foreground (main part of the crane) and background (other objects). Hence, to decrease the possibility of misclassification. For this purpose, different approaches which are mostly based on segmentation can be used (Kaur & Kaur, 2014).

Our proposal for deriving foreground-background separation is to take advantage of the YCrCb colour space properties applying first a transformation on HSV colour space, followed by morphological operations and connected components analysis (Jähne, 2002). In YCrCb, Y is the luminance obtained from RGB after gamma correction; Cr is the chromatic red represents how far is the red component from the luminance; and Cb is the chromatic blue and represents how far is the blue component from the luminance. The YCrCb colour space can be derived from the RGB colour space (Ford & Roberts, 1998):

$$\begin{bmatrix} Y\\ C_R\\ C_B \end{bmatrix} = \begin{bmatrix} \frac{K_R}{2} & \frac{K_G}{1-K_B} & \frac{K_G}{2} & \frac{K_B}{1-K_B} \\ \frac{1}{2} & -\frac{1}{2}\frac{K_G}{1-K_R} & -\frac{1}{2}\frac{K_G}{1-K_R} \end{bmatrix} \begin{bmatrix} R\\ B \end{bmatrix}$$
(1)

 $K_R = 0.299$ $K_G = 0.587$ $K_B = 0.114$ where

This colour space separates the luminance and chrominance components into separate, uncorrelated channels. It can be interpreted as a normalisation in intensity of R and B channels of RGB colour space (García-Mateos et al., 2015). Therefore, the behaviour of the chromatic part can be analysed separately from the luminance.

In Figure 3.a the original image histogram of chromatic blue channel is analysed; it can be seen that the Cb channel has two well-defined peaks. This would allow making a thresholdbased operation which is useful to distinguish between the crane (the foreground) and other objects, because this pattern is repeated along the images (with some small variations).

In order to make the cranes' blue colour more distinct, an empirical transformation in HSV colour space is applied. First, the hue is rotated by an angle of 30° , and then the saturation multiplied by four. This is visible in the 3D histogram in Figure 3.b, and e, where the colours expand in opposite directions. The advantage of applying this transformation is clearly visible in Figure 3.f. This is useful for the foregroundbackground separation, because a fixed threshold can be defined independent of the image.



Figure 3. HSV transformation. First row relates to images before HSV transformation and second row shows their counterparts after HSV transformation. a, d) RGB image, b, e) 3D YCrCb histogram; c, f) histogram of the Cb channel of a, and d.

Once the transformation is done, the image is again converted to YCrCb colour space and an optimal empirical threshold value is applied. After employing morphological cleaning, a connected component operator is applied to get separated segments. Then, segments areas are checked to obtain the biggest segment. Exploiting the Euler number as a measure for topologic properties, holes in this segment are filled and the area of interest is extracted.

2.2.2 Classification: Pixel-based classification is used to separate the defects and colour changes from other parts of the cranes. Bayes classifier is one well-established classifications approach. This classifier uses the a priori probabilities of the

events to estimate the probability of unseen events by means of the Bayes' theorem (cf. Equation (2)). It also uses historical or training data to calculate the observed probability of each event as a function of its characteristic vector. In general, to make a prediction, the query set is used. The same features as in the training process are feed to the classifier, and as output some probabilities are obtained, from which it is possible to estimate the most likely class. In the validation phase, predicted class is compared with the true label to evaluate the overall performance of this classifier.

$$p(class_i|\mathbf{x}) = \frac{p(\mathbf{x}|class_i) p(class_i)}{p(\mathbf{x})}$$
(2)

The *evidence* can be represented as (3) and is the probability of finding a pixel with certain feature vector \mathbf{x} in the image, from any class, that is, the probability that any pixel on the image takes those values. This is a normalising factor and could be used when the *argmax* is not the final decision-making operator.

$$p(\mathbf{x}) = \sum_{1}^{M} p(\mathbf{x}|class_i) p(class_i)$$
(3)

The *likelihood* or conditional probability describes the chances of finding a pixel with certain feature vector \mathbf{x} from each of the possible classes. Those conditional probability functions are estimated from labelled training data for each class. An example is illustrated in Figure 4.



Figure 4. Likelihood representation of some classes

The *priors* define the probability that pixels from a certain class appear in the images and the *posteriors* define the probability that a pixel belongs to a certain class. A set of possible classes is previously defined, according to our needs. The main objective is to classify each pixel into the most likely class. Hence, according to Richards (2013), the rule to decide if a pixel belongs to a certain class, will be if the likelihood belonging to that class is bigger than any other likelihood of that pixel belonging to another class. This can be expressed as follows:

$$\boldsymbol{x} \in class_i \leftrightarrow p(class_i | \boldsymbol{x}) > p(class_j | \boldsymbol{x}) \forall j \neq i$$
(4)

For an N dimensional space, the specific form of the Gaussian multivariate normal likelihood distribution is defined as follows:

$$p(\mathbf{x}|class_{i}) = \frac{e^{-\frac{1}{2}((\mathbf{x}-\mathbf{m}_{i})^{T}\mathbf{C}_{i}^{-1}(\mathbf{x}-\mathbf{m}_{i})}}{\frac{N/2}{\sqrt{2\pi}\sqrt{|\mathbf{C}_{i}|}}}$$
(5)

The terms which do not depend on class values can be removed, and when there is no useful information about the values of the prior probabilities they are assumed to be equal (Richards, 2013), and this fact defines as (6) the general form of the Gaussian maximum likelihood classifier, better known as Gaussian Naïve Bayes classifier which determines the class membership of a pixel based on the highest class conditional probabilities, or likelihoods..

$$g_i(\boldsymbol{x}) = -ln|\boldsymbol{\mathcal{C}}_i| - (\boldsymbol{x} - \boldsymbol{m}_i)^T \boldsymbol{\mathcal{C}}_i^{-1} (\boldsymbol{x} - \boldsymbol{m}_i)$$
(6)

where m_i : mean vector of the data in class_i C_i : covariance matrix of the data in class_i.

In Figure 5 a sample of the colour histogram and its Gaussian approximation is depicted.



Figure 5. Histograms of one class and fitted Gaussians

Some practical considerations must be taken into account, in order to represent the classes in the training data properly. For training the classifiers, representative data of the problem is needed. In order to check if the selected algorithms would work for our purpose, first, some crane images have been taken from the image database. Afterwards, some image patches are manually selected and labelled. In this approach, pixels will be classified into one of the predefined number of classes, even if the probability is small because the Gaussian distribution takes values on the full domain. Therefore, it is important to select a proper optimal number of classes: bad results can be obtained if some of them are overlooked. Moreover, to be able to estimate the parameters of their distributions, enough training data for each class needs to be provided.

The probability distribution of the selected classes is depicted in Figure 6; the whole spectrum is covered, but there is a range covered by the tails of probability distribution functions for all the classes, this is due to inter-class similarity.



Figure 6. Gaussian distributions of the classes

2.3 Database management

In this project, it is expected that large image datasets using drone technology be captured. When working with large drone-based image datasets, which are captured for several objects (here different cranes) and in multiple epochs, the data management can be a challenge and should therefore be carefully planned. Drone-based images need to be preprocessed before they can be used efficiently and this has an impact on the management process and should be considered before the data management design is implemented. The preprocessing steps lead to an accumulation of data which grows in complexity and which needs to be stored correctly (Huang et al., 2018). An efficient way to store the captured images is to use a database. It must be designed in a way, that not only the original images and the final results are integrated but also the intermediate results should be considered. In addition, the data management concept should be flexible enough to adapt to non-predictable situations which might occur during the ongoing development.

In the following the structure and storage of the images in the file system is shortly discussed. Next, linking metadata in the database with the images in the file system is explained. The section concludes with a short presentation of the current state of the snowflake-based table implementation in the database.

2.3.1 Data structure and storage: The images captured by the drone are transferred to a data server and stored directly into the file system. It is expected that all images contain additional metadata with important information which are necessary for the processing of the images. For example, the metadata includes the date- and timestamp when the image was captured, information about the camera settings, and the geographic position of the centre of the image.

The storage of the individual images in the file system follows a predefined structure. This guarantees that a distinct image can be accessed without any delay. In addition to the structure of the file system, the image name should be unique to clearly identify each image. They are therefore re-named automatically based on specific metadata (consisting both EXIF metadata e.g. the image capture date and time plus additional non-EXIF metadata such as the name of neuralgic point they are showing e.g. A1, A2) using a Python-based tool developed for this task to avoid the need of manual re-naming of each image separately. Furthermore, the new names of the images are composed of the additional non-EXIF metadata like crane type e.g. identifying the vendor, the crane bridges e.g. CB1, and the direction from which the photo was taken e.g. Right and Left (referring to a viewing direction along the quay).

2.3.2 Database Concept: Particularly, to store large dronebased image datasets a combination of storing the image itself on the file system and the attached metadata into a database management system could be used (Fan et al., 2017). This approach has the advantage that the images are not stored directly in the database and therefore cannot be responsible for performance issues. The images stored in the file system are only linked to the database indirectly by adding the physical path as an attribute to the table which contains the metadata. This concept guarantees that the volume of the data inside the database does not become too extensive and that the data can be queried efficiently.

I order to implement the approach described above, it is here suggested to use the object relational database system PostgreSQL. PostgreSQL is an open source product with a wide selection of available features. The PostGIS extension can be installed additionally to transform the relational PostgreSQL database into a spatial database. This adds support for database records with an attached geometry and furthermore allows to query the data based on spatial indices which accelerates the process of receiving the needed data. In addition, the PostGIS extension adds support for vector and raster data (Marquez, 2015). The PostgreSQL database in combination with PostGIS is also highly compatible with other open source software products which are dealing with data visualization and processing. Namely the QGIS and GeoServer (Alamouri & Gerke, 2019). Especially the latter can become relevant later in the project, when the final results are presented to the user via a web interface.

To avoid data redundancy, dependency and anomalies during the data entry, update, deletion in the database a well-known technique of database normalization is considered and realised for database schema design and implementation.

The earliest articles written on database normalization appeared during 1970s. For example, you may refer to Codd (1971) as one of the oldest database schemas. The database normalization is a vast and well researched topic but, in brief, the main idea of the database normalization is to prevent data duplication inside database tables/columns which ensures the data consistency and integrity and ultimately results in utilizing less space for storage. Besides, it makes the database more flexible.

To implement the concept of normalization in database schema within the context of this research project, a database schema model known as *snowflake* schema (Figure 7), which is a variant of another schema called star-schema, is used (van der Lans, 2012).



Figure 7. A Snowflake schema diagram

In a snowflake schema model, a main table which is referred to "Fact Table" is created. This table is connected to other tables which are referred as "Dimension Tables". A fact table in snowflake schema has only one relationship with dimension tables. In regards to current application, dimension tables may be individual tables to store crane details e.g. name, bridges, neuralgic points and etc and then these tables can be referenced in a crane table that will act as fact table. This gives a multidimensional and hierarchy style schema (van der Lans, 2012). Each dimension stores particular data which is not directly relevant to other dimensions, hence, modifying, and deleting of entries in one dimension does not affect other dimensions. This particular schema model enables normalizing data in the database by using fact and dimension tables.

3. PRELIMINARY RESULTS

At HHLA's Container Terminal Tollerort (CTT), where we implement and test our system, a total of 14 container gantry cranes of different years of construction from various manufacturers are in operation (Figure 8). In order to maintain the uninterrupted and all-day operation of the container gantry cranes (24 hours/ 365 days) for the loading and unloading of container ships in a seaport, the performance and evaluation of a qualified inspection is extremely important.



Figure 8. Schematic representation of one crane with 13 critical/neuralgic points

3.1 Image analysis results

3.1.1 Foreground-background separation: Some of the results of the previous explained foreground-background separation method are shown in Figure 9. Some samples of neuralgic area images have been evaluated visually, and as it can be seen, the method selects the biggest segment and crops it with a good performance.

3.1.2 Classification results and evaluation: The Naïve Bayes classifier is trained with a total of 400 image patches which contain 1M training pixels using a total number of 5 predefined classes (crane blue, crane red, crane repaint, rust and metal bar). The quality measures that are commonly used in the literature for classification purposes are precision, recall and f1-score, which are defined as follows (Géron, 2019):

$$Precision = \frac{TP}{TP + FP}$$
(7)

$$Recall = \frac{TP}{TP + FN}$$
(8)

$$f1_score = 2 \frac{precision \times recall}{precision + recall}$$
 (9)





The results of the Naïve Bayes classifier trained with Y, Cr and Cb features are shown in Table 2. Overall weighted accuracy and recall are 80% and 83%, respectively.

	Precision	Recall	f1-score
Crane_blue	0.74	0.90	0.81
Crane_red	0.99	0.97	0.98
Crane_repaint	0.35	0.14	0.20
Metal_bar	0.54	0.98	0.69
Rust	0.99	0.87	0.93
Weighted avg.	0.80	0.83	0.81

Table 2. Results on test set of Naïve Bayes classifier

As, in the current phase of the project, the target class is *rust*, we analyse the result of this class, separately and all other classes serve as negative class. The high value of precision and recall and consequently high f1-score are promising. However, because of the nature of the target class (rust) and dangerous consequences of missing some rust areas, recall measure should be our focus in future steps of the project, even at the cost of having lower precision. The preliminary results on the validation set images are shown in Figure 10.



Figure 10. Preliminary results on validation set. a) Original image, b) Prediction of Naïve Bayes classifier

Most parts of the colour changes are detected (in red) successfully. However, the method has problems in distinguishing some rusts crane red class and also crane blue

from crane repaint class. Both these issues are on account of the fact that inter-class similarities of these classes in the current feature space are high.

3.2 Data management results

A database preliminary schema model based on previously explained snowflake schema concept is developed for this research project.

As discussed in section 2.3, the input data (images), the output of image analysis process and the annotations provided by the user/operator are saved in the structure of the file system on the server and their storage path is referenced in relevant database tables. This schema model ensures the database integrity and data normalization. However, the challenge that may be faced is the adding more joins in query of data which with the developments in Relational Database Management Systems (RDMBS) is minimized to some extent.



Figure 11. Database schema/structure

In Figure 11 the main fact table is the *crane_table* which is connected to other tables or dimension e.g. *crane_type_table*, *neuralgic_points*, etc. using their primary keys. The primary keys of dimensions are stored as foreign keys and attribute inside the fact table. The dimensions may contain their own attributes and sub-dimension e.g. *epoch_table* but cannot be connected to each other.

4. CONCLUSIONS

Multi-epoch structural health monitoring is the ultimate aim of our research project. In this paper the general concept and preliminary results of various modules of single-epoch solution are presented. Complex objects' structure and difficulties in accessing the parts of interest of the objects, necessitates a cautious flight planning and data capturing. Moreover, data management is a key challenge of the project due to the large number of images (more than 10 cranes which are captured at least 4 times per year and having 500 images from each crane per epoch). Hence, a file system-based data storage is utilized for managing the images and associated attributes. The images in the file system are linked to PostgreSQL/PostGIS based databases indirectly by adding the physical path as an attribute to the tables which contains the metadata of the images.

A machine learning approach based on Naïve Bayes classifier is employed to separate the target class from other parts of the objects in the image. The preliminary results indicate, given that enough representative training data from representative classes are provided, that the pixel-based classification can be employed to separate the defects (here rust is considered) with a very high precision for this class. However, 87% percent recall in this class is not enough for this application and we should improve our approach to have higher recall value, even at the cost of lower precision. While having a defect of size one pixel is very unlikely and misclassification of individual pixels is not important in practice, object-based approaches could be promising to increase the reliability of the detection system. Moreover, ensemble learning could also help to control the capability of the system to recall the possible defect reliable. Exploiting geometric information such as camera poses in local (crane) coordinate system is another possibility to restrict the search space closer to the area of interests. Advanced relationships between geospatial data and database tables should also be investigated. Last but not least taking advantage of state-of-the-art deep learning approaches is pending in our project mainly due to the current lack of training samples.

ACKNOWLEDGEMENTS

This project is part of the Research Program "Innovative Port Technologies" (IHATEC) which is funded by the German Federal Ministry of Transport and Digital Infrastructure (BMVI). The authors would like to thanks HHLA Sky and HHLA CTT for their supports.

REFERENCES

Alamouri, A., & Gerke, M. (2019). Development Of a Geodatabase for Efficient Remote Sensing Data Management in Emergency Scenario. ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences, IV-2/W5(2/W5), 87–93. https://doi.org/10.5194/isprs-annals-IV-2-W5-87-2019

Blaschke, T., Hay, G. J., Kelly, M., Lang, S., Hofmann, P., Addink, E., Queiroz Feitosa, R., van der Meer, F., van der Werff, H., van Coillie, F., & Tiede, D. (2014). Geographic Object-Based Image Analysis – Towards a new paradigm. ISPRS Journal of Photogrammetry and Remote Sensing, 87, 180–191. https://doi.org/10.1016/j.isprsjprs.2013.09.014

Codd, E. F. (1971). Normalized data base structure. Proceedings of the 1971 ACM SIGFIDET (Now SIGMOD) Workshop on Data Description, Access and Control -SIGFIDET '71. https://doi.org/10.1145/1734714.1734716

Fan, J., Yan, J., Ma, Y., & Wang, L. (2017). Big Data Integration in Remote Sensing across a Distributed Metadata-Based Spatial Infrastructure. Remote Sensing, 10(2), 7. https://doi.org/10.3390/rs10010007

Ford, A., & Roberts, A. (1998). Colour Space Conversions. http://www.inforamp.net/

García-Mateos, G., Hernández-Hernández, J. L., Escarabajal-Henarejos, D., Jaén-Terrones, S., & Molina-Martínez, J. M. (2015). Study and comparison of color models for automatic image analysis in irrigation management applications. Agricultural Water Management. https://doi.org/10.1016/j.agwat.2014.08.010

Géron, A. (2019). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition [Book] (N. Tache (ed.); 2nd ed.). O'Reilly. https://www.oreilly.com/library/view/hands-on-machinelearning/9781492032632/

Hoskere, V., Narazaki, Y., Hoang, T. A., & Spencer, B. F. (2020). MaDnet: multi-task semantic segmentation of multiple types of structural materials and damage in images of civil infrastructure. Journal of Civil Structural Health Monitoring. https://doi.org/10.1007/s13349-020-00409-0

Huang, Y., Chen, Z. X., Yu, T., Huang, X. Z., & Gu, X. F. (2018). Agricultural remote sensing big data: Management and applications. Journal of Integrative Agriculture, 17(9), 1915–1931. https://doi.org/10.1016/S2095-3119(17)61859-8

Jähne, B. (2002). Digital Image Processing, 5th revised and extended edition. Measurement Science and Technology. https://doi.org/10.1088/0957-0233/13/9/711

Kaur, D., & Kaur, Y. (2014). International Journal of Computer Science and Mobile Computing Various Image Segmentation Techniques: A Review. In International Journal of Computer Science and Mobile Computing.

Kerle, N., Nex, F., Gerke, M., Duarte, D., & Vetrivel, A. (2019). UAV-Based Structural Damage Mapping: A Review.

ISPRS International Journal of Geo-Information, 9(1), 14. https://doi.org/10.3390/ijgi9010014

Maboudi, M., Amini, J., Hahn, M., & Saati, M. (2017). Objectbased road extraction from satellite images using ant colony optimization. International Journal of Remote Sensing, 38(1), 179–198. https://doi.org/10.1080/01431161.2016.1264026

Marquez, A. (2015). PostGIS Essentials. Packt Publishing Ltd. https://www.packtpub.com/product/postgisessentials/9781784395292

Nex, F., & Remondino, F. (2014). UAV for 3D mapping applications: A review. Applied Geomatics, 6(1), 1–15. https://doi.org/10.1007/s12518-013-0120-x

Rao, A. S., Nguyen, T., Palaniswami, M., & Ngo, T. (2020). Vision-based automated crack detection using convolutional neural networks for condition assessment of infrastructure. Structural Health Monitoring. https://doi.org/10.1177/1475921720965445

Richards, J. A. (2013). Remote sensing digital image analysis: An introduction. In Remote Sensing Digital Image Analysis: An Introduction. https://doi.org/10.1007/978-3-642-30062-2

Sahebdivani, S., Arefi, H., & Maboudi, M. (2020). Rail Track Detection and Projection-Based 3D Modeling from UAV Point Cloud. Sensors, 20(18), 5220. https://doi.org/10.3390/s20185220

Saleem, M. R., Park, J. W., Lee, J. H., Jung, H. J., & Sarwar, M. Z. (2020). Instant bridge visual inspection using an unmanned aerial vehicle by image capturing and geo-tagging system and deep convolutional neural network. Structural Health Monitoring.

https://doi.org/10.1177/1475921720932384

Stein, M. (2018). Conducting Safety Inspections of Container Gantry Cranes Using Unmanned Aerial Vehicles (pp. 154– 161). Springer, Cham. https://doi.org/10.1007/978-3-319-74225-0_20

van der Lans, R. F. (2012). Business Intelligence and Data Warehousing. In Data Virtualization for Business Intelligence Systems (1st Editio, pp. 27–57). Elsevier. https://doi.org/10.1016/b978-0-12-394425-2.00002-2