Autonomous Intelligent Systems

REVIEW Open Access

Machine learning techniques for robotic and autonomous inspection of mechanical systems and civil infrastructure

Michael O. Macaulay[1]* and Mahmood Shafiee[1]

Abstract

Machine learning and in particular *deep learning* techniques have demonstrated the most efficacy in training, learning, analyzing, and modelling large complex structured and unstructured datasets. These techniques have recently been commonly deployed in different industries to support robotic and autonomous system (RAS) requirements and applications ranging from planning and navigation to machine vision and robot manipulation in complex environments. This paper reviews the state-of-the-art with regard to RAS technologies (including unmanned marine robot systems, unmanned ground robot systems, climbing and crawler robots, unmanned aerial vehicles, and space robot systems) and their application for the inspection and monitoring of mechanical systems and civil infrastructure. We explore various types of data provided by such systems and the analytical techniques being adopted to process and analyze these data. This paper provides a brief overview of machine learning and deep learning techniques, and more importantly, a classification of the literature which have reported the deployment of such techniques for RAS-based inspection and monitoring of utility pipelines, wind turbines, aircrafts, power lines, pressure vessels, bridges, etc. Our research provides documented information on the use of advanced data-driven technologies in the analysis of critical assets and examines the main challenges to the applications of such technologies in the industry.

Keywords: Machine learning, Deep learning, Robotics and autonomous system (RAS), Inspection, Monitoring, Mechanical systems, Civil infrastructure

1 Introduction

There has been considerable literature concerning the deterioration of critical systems and infrastructure around the world, and the resulting health and safety implications, whether these are roads, bridges, or energy related infrastructure. As reported by [1], there are at least 150,000 bridges in the United States alone that have lost their structural integrity and are no longer fit for purpose. Mechanical systems and civil infrastructure deemed critical assets by both government and industry, are vulnerable to damage mechanisms, which can adversely affect social services and the overall productivity of an economy.

This has ensured that regular inspection and maintenance is now standard practice. The operation and maintenance (O&M) costs, resulting from standardized inspection and maintenance practices, have been quite considerable for government and industry. O&M cost accounts for a large proportion of lifecycle costs in critical systems; for instance, the O&M expenditure in the wind energy industry amounts to 25%-30% of total costs [2]. Challenges to conventional maintenance and inspection practices of civil infrastructure and mechanical systems involves the fact that most methods and protocols employed are bureaucratic and labour intensive. The in-

*Correspondence: macaulay.michael@gmail.com; mm2168@kent.ac.uk
[1] Mechanical Engineering Group, School of Engineering, University of Kent, Canterbury CT2 7NT, United Kingdom

spection and monitoring of assets are usually undertaken manually, with technicians and operators having to travel sometimes to distant locations hundreds of miles away. In some cases, operators and technicians must work in environments which are subject to intensive conditions caused by heat, cold, noise, wetness, dryness, etc. In other cases, the location may be inaccessible to human technicians, as in the case of large storage tanks or underground pipelines.

Technological advancements and emergence of robotics and autonomous systems (RAS) have begun to revolutionize monitoring and inspection of mechanical systems and civil infrastructure. This revolution has provided an interest and demand for the use of RAS technologies to support the monitoring, inspection and maintenance of offshore wind farms, gas and utility pipelines, power lines, bridges, railways, high rise buildings, vessels, storage tanks, underwater infrastructure, etc., in order to mitigate the current health and safety risks that human operators currently experience while inspecting or monitoring such infrastructure within the energy, transport, aerospace and manufacturing sectors [3]. There is a drive in both industry and government for the development and availability of RAS technologies that can be deployed to provide data on the condition of assets and help technicians undertake actions deemed necessary, based on the information provided by RAS. This information can be signals provided by hardware instruments, or images taken by cameras from damaged, shadowed, rough, or rusty surfaces.

Robot inspections have been proven to be more efficient and faster than human inspections. For instance, the inspection of wind turbines using unmanned aerial vehicles (UAVs) take considerably shorter time than that using conventional visual inspection [3]. As indicated in a case study reported by [4], the traditional rope access method can inspect only one wind turbine per day, whereas a UAV can inspect up to three wind turbines in a day. Vast amount of data with diverse formats (such as, audio, video, or digital codes) can be collected by numerous RAS technologies that are deployed to monitor and inspect infrastructure. However, it will be quite time-consuming, if not impossible for human operators to analyze this volume of incoming data using conventional computing models. Machine learning (ML) techniques provide advanced computational tools to process and analyze all the data provided by RAS technologies efficiently, speedily, and accurately. The evolution from teleoperated robot systems which require remote human control to autonomous systems, which when pre-programmed can operate without human intervention has helped in the maturity and ascendance of RAS technologies which remove the need for travel, bureaucratic paperwork requirements, etc. While there might still be a number of RAS technologies that work offline, there is a growing number of RAS technologies that are wireless, remotely transferring data, e.g., images of structures, materials, etc. to a control office through inter-networks for analysis.

The aim of this paper is to provide an academic contribution by reporting on literature and research related to the use of ML in RAS-based inspection and monitoring of mechanical systems and civil infrastructure. It also proposes a classification and analysis of different ML techniques used for the analysis of data yielded from RAS-based inspections. This means that the research in this paper investigates and identifies which study, in which literature, has used which ML technique, to support different RAS technologies deployed for inspection purposes. To achieve this aim, we identify the relevant literature with keywords including: robotics, inspection, machine learning, maintenance, mechanical engineering, civil infrastructure, and asset. We will also provide a review and classification of ML techniques; the types of damage mechanisms being considered, e.g., corrosion, erosion, fatigue, cracks, etc.; the types of inspections; and robotic platforms that have been used to support both industry and academic research. In addition, a review will be conducted on the characteristics of datasets collected during RAS inspections of civil and mechanical infrastructure, including: sources of data (public or non-public); types of data (e.g., image, video, documents etc.); size of data; velocity or rate of data generation and transmission; and the variety of data (structured or unstructured). Following on from this, there will be an evaluation of the results and findings. Finally, there will an exploration of potential development in RAS for inspection and monitoring of future assets.

The rest of this paper is organized as follows. Section 2 reviews different types of RAS technologies that have been proposed and designed to support the inspection and monitoring of mechanical systems and civil infrastructure. Section 3 reviews the characteristics of the data collected by RAS systems for inspection and monitoring purposes. Section 4 reviews various types of ML techniques that can and have been used to process and analyze data from RAS inspections. Section 5 discusses the findings of the literature review undertaken in this research and then finally, Section 6 reviews some of the current technology gaps and challenges in the application of ML techniques for RAS based inspection of mechanical systems and civil infrastructure. The organization of this literature review is schematically illustrated in Fig. 1.

2 RAS technologies for monitoring and inspection

Today, there are a variety of robotic and autonomous systems being developed and deployed in various industries, including aerospace, manufacturing, energy, transport, agriculture, healthcare, etc. RAS systems are widely used to support monitoring, maintenance and inspection of mechanical systems and civil infrastructure. These technologies are provided with artificial intelligence (AI),

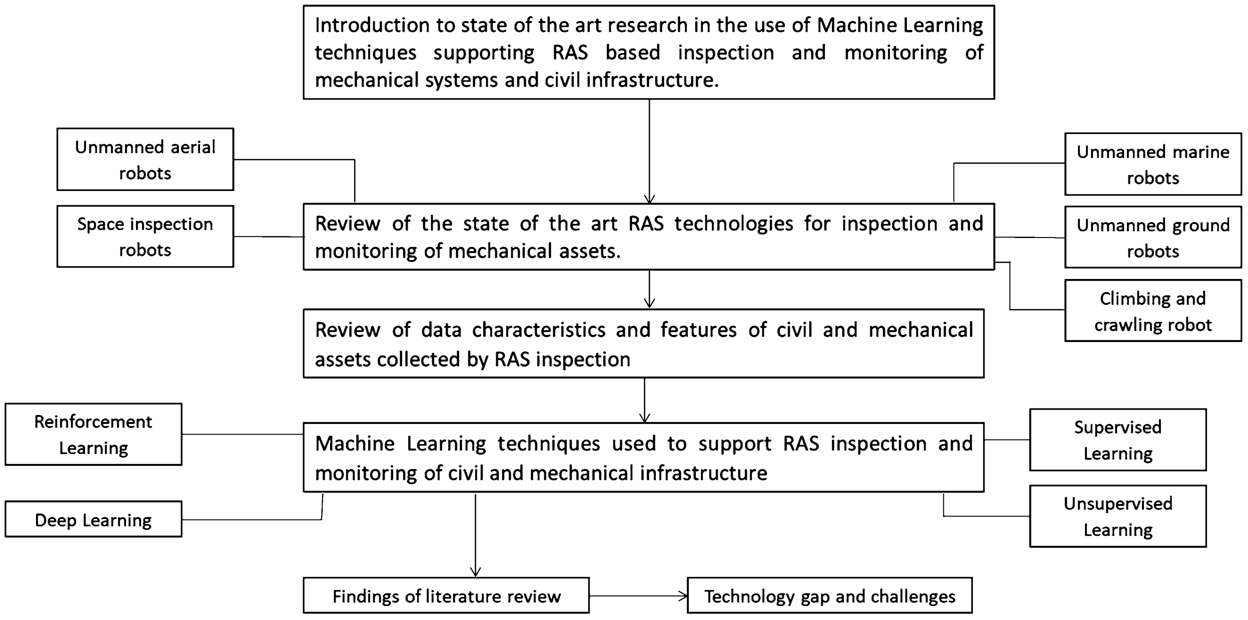
Introduction to state of the art research in the use of Machine Learning techniques supporting RAS based inspection and monitoring of mechanical systems and civil infrastructure.
Unmanned aerial robots
Space inspection robots
Review of the state of the art RAS technologies for inspection and monitoring of mechanical assets.
Unmanned marine robots
Unmanned ground robots
Climbing and crawling robot
Review of data characteristics and features of civil and mechanical assets collected by RAS inspection
Reinforcement Learning
Deep Learning
Machine Learning techniques used to support RAS inspection and monitoring of civil and mechanical infrastructure
Supervised Learning
Unsupervised Learning
Findings of literature review
Technology gap and challenges

Figure 1 Schematic illustration of organization for the literature review

sometimes referred to as machine learning (ML), to enable and complete complex tasks, as well as process vast amounts of data. The mechanical design for an RAS system used to support inspection, monitoring and maintenance purposes, can be categorized by specific locomotion and adhesion mechanisms. The adoption of an inspection robot's locomotion and adhesion mechanisms is sometimes offset against task or application specific requirements such as payload, power requirements, velocity, and mobility [5, 6].

Locomotion in robotics specifically refers to directional movement that makes it possible for an object to move from location to another, the mechanism that makes a robot capable of moving in its environment. The literature states that there are four main types of locomotion that a robot system could be fitted with, depending on the task and environment they are being built to support [5]. These four locomotion types include: arms and legs, wheels and chains, sliding frame, and wires and rails. Considering the pros and cons of each type, arm and legged robots are better suited for maneuver around obstacles in the environment when compared to other locomotion systems. Conversely, wheels and chain-driven locomotion are best suited to environments with a flat and even surface and ill-suited for navigating obstacles in the environment. The sliding frame locomotion mechanism comprises of a mechanical design that has two frames which move against one another in rotation. This mechanical design however provides for low speeds. Finally, locomotion involving wires and rails comprises of a simple system where the robot is held in place by wires and rails [5]. Adhesion in robotics refers to the mechanism by which robot systems can attach or cling to surfaces in their environment. Common adhesion mechanisms for RAS systems that support inspection tasks ranging from magnetic adhesion and pneumatic adhesion (whether that be the passive suction cups type or active suction chambers type or vortex thrust systems type) to vacuum sucker, propeller, and dry adhesion.

In the following subsections, we review RAS-based inspection systems that are currently being used in different industry sectors. These systems range from platforms operating below sea level to those operating within the troposphere (ground level to about 10-20 km above sea level) and the ones purposed to operate in the thermosphere and exosphere (space and beyond). We therefore suggest five robot categories, including: unmanned marine robots, ground-based robots, climbing and crawler inspection robots, unmanned aerial robots, and space inspection robot systems.

2.1 Unmanned marine robot systems

Some literature use the term 'unmanned marine vehicles (UMV)' as an umbrella term for unmanned surface vehicles (USV) and unmanned underwater vehicles (UUV) respectively [6]. UUVs however can be further classified as either autonomous underwater vehicles (AUV) or remotely operated vehicles (ROV) [7].

AUVs are unmanned, pre-programmed robot vehicles, purposed and deployed into the ocean depth, autonomously without the support of cabling and human intervention. When the AUV completes its task, it returns to a pre-programmed location, where its data can be retrieved, downloaded, processed, and analyzed. An ROV is an unmanned robot which is deployed into ocean depths; however, the difference is that an ROV is connected to a ship by cables. An operator located on the ship pilots the ROV. The cables attached to an ROV are used to transmit commands and data between the operator and robot. AUVs can be deployed to support inspection of hazardous objects, surveying and mapping wrecks, and deep underwater infrastructure (e.g., subsea cables). ROVs are usually deployed into dangerous or challenging deep water environments for human divers. Therefore, both AUV and ROV robot systems are fitted with and supported by a variety of sensors to collect data. The data provided may be used for military or civilian surveys, inspections, surveillance, and exploration purposes. AUVs and ROVs are usually equipped with cameras for obtaining video images underwater. ROVs use cameras to transmit video telemetry to human operators for analysis and decision-making. Sound navigation ranging (sonar) and fiber optic gyros (FOG) support object detection, obstacle avoidance and navigation. ROVs might also be fitted with robotic arms for collecting underwater samples [8].

2.2 Unmanned ground robot systems

Unmanned ground-based robots operate autonomously on ground surfaces. They are sometimes referred to as mobile robots in some academic literature, alternatively they are otherwise referred to as unmanned ground vehicles (UGV) or land-based robots. Ground-based robots in literature are also sometimes categorized based on their locomotion, which among other criteria is based on the environment it is deployed into, which are usually even, stable environments. A ground-based robot has the advantage of being able to support and carry maximum amounts of payload where appropriate, however the disadvantage of these robot types is their lack of mobility with uneven terrain [6].

Unmanned ground-based robots can be categorized as wheeled robots, walking (or legged) robots, tracked robots, or robots with a hybrid of either wheeled, legged or tracking [9]. Wheeled robots navigate on the ground using motorized wheels to propel themselves [9]. Literature states that there are four types of wheeled robot, which can be differentiated by the number of degrees of freedom (DOF) they hold. DOF is defined as the number of independent variables that can define the motion or position of an object (or mechanism) in space. These four types include the fixed standard wheel; the castor wheel; Swedish wheel, and finally the ball or spherical wheel. There are also several types of wheeled robots, including the single-wheeled robot, two-wheeled robot, three-wheeled robot, and so on, each with their unique mobility feature or characteristic [9].

Legged (or walking) robots, unlike wheeled robots, navigate on both even and uneven surfaces, hard and soft surfaces, and can detect obstacles in their path or environment. Legged robots can be classified as one-legged (hoppers), two-legged (humanoid), three-legged, four legged (quadruped), five-legged, six-legged (hexapod), and so on [9]. Hybrid ground-based robots are robots that combine legged, wheel and track locomotion systems in any given configuration [9].

The applications of unmanned ground robots are numerous, and they range from the nuclear industry, where human operators are replaced by robots to operate in radioactive environments, to military operations for surface repairs, navigating minefields, explosive ordinance disposal (EOD), carrying and transporting payload, etc. Other state-of-the-art applications include reconnaissance, surveillance, and target acquisition operations; and space exploration as in the case of NASA's planetary rovers [10]. Unmanned ground-based vehicles are also fitted with array of sensor payload options to support autonomous operations, navigation through the environment and data collection. Cameras are used to scan the robot's environment and support calculation of its position. Furthermore, motion detectors, infrared (IR) sensors, temperature and contact sensors support object detection, obstacle and collision avoidance and obstacle localization. Laser range finder sensors which use a laser beam to generate distance measurements, producing range data, also support object detection and obstacle avoidance [8].

2.3 Climbing and crawler inspection robots

Wall climbing and crawler robots were developed for movement on vertical plane environments for the inspection and maintenance of a range of assets such as storage tanks, nuclear power facilities, and high-rise buildings [11, 12]. Oil refineries consist of storage tanks that require cleaning; along with a requirement for routine inspection and non-destructive testing (NDT) of these tanks to check for cracks and leaks. The traditional and manual implementation of these routine inspection and maintenance tasks results in very high labor and financial costs. The development and deployment of climbing and crawling robot systems can help with automating these tasks [12].

Climbing robots adopt an adhesion mechanism, based on the type of the environment they are deployed into [6]. These robots employ magnetic adhesion or pneumatic or negative pressure (depending on the suction or thrust type) adhesion. According to literature, climbing robot systems are usually fitted with arm and leg locomotion mechanism systems. The number of arms and legs can vary from two to eight legs, albeit eight legged robots are not as common. Alternatively, climbing and crawler robots can otherwise be fitted with wheels of chain-driven locomotion [5]. The adhesion mechanisms available today make climbing and crawling robots capable of attaching to structures and materials, while also providing a reliable platform for attached payload and tools [6].

Literature reports some information about the velocity, and mobility of the climbing and crawler robots. Climbing robot systems might need to reach high velocity on a vertical plane, for optimal movement between inspection locations. With respect to mobility, climbing robot systems which are fitted with arms and legs can navigate uneven surface, steps, and other objects in the environment [5]. When considering the payload requirements for climbing and crawler robots, some sensors such as ultrasonic sensors, gravity sensors, acceleration sensors etc. are used to measure and provide data about the distance of objects or obstacles in front [5, 6, 11–13]. The literature provides a varied advice on the weight of the payload capacity, ranging between 10 kg and 30 kg. Obviously, the weight requirements will be dependent on the tasks that the robot system is deployed to complete [5, 11, 12]. However, robotics engineers must engage with the problem that climbing and crawling robots will need to suitably offset the take up of heavy payloads, while securing adhesion on the challenging surfaces that they are designed for [6].

2.4 Unmanned aerial robot systems (or drones)

Unmanned aerial robot systems are interchangeably referred to as unmanned aerial vehicles (UAVs), and more commonly referred to as drones in the literature. Various multidisciplinary disciplines, ranging from environmental monitoring to civil engineering are increasingly deploying drones to support with various inspection type applications [14]. This is because research has continuously shown that the use of drones for inspection-based tasks reduce the need for human actors, their risk and exposure to injury and fatality, maintenance, and downtime costs [14–16]. Seo *et al.* [14] indicates that the selection of a drone for a particular application is based on some criteria including the mission duration, battery life, camera and video resolution, payload capacity, GPS and collision avoidance and cost performance [14]. Locomotion in drones is provided by propellers, or as referred to rotors in certain literature. The term propellers and rotors are used interchangeably in the literature, although technically, a rotor could be considered a horizontal propeller (such as those mounted on a helicopter), while propeller could technically be the vertical rotor mounted on an airplane. Nevertheless, they are identical objects seen from different angles. A propeller propels an object, using the thrust as the force for horizontal movement and lift for vertical movement – providing for a vertical take-off and landing (VTOL) capability. Motors provide power to propellers and spin them at high speeds. These high speeds in turn create the thrust or lift which provide the drone with the required locomotion [15]. Lattanzi *et al.* [6] contended that drones compromise with having reduced stability and how much payload they can support, with the advantage of increased movement and mobility [6]. In literature, flight time or duration of operation is inextricably linked to battery life or number of batteries available in a drone. The battery provides the electricity to power the drone, and most drones are only capable of providing enough power to support 20-30 minutes of flight time [14].

Drones are described in literature as technology platforms that can support a variety of applications and carry a variety of sensor payloads. Sensor payloads can vary from thermal and infrared to optical cameras. Light detection and ranging (LIDAR) is used to measure and provide data on the distance of objects, and with radar, the angle and velocity of objects as well [16]. Literature indicates that the higher the number of propellers (or rotors) supporting the drone, the greater the drone's payload capacity. Although there are caveats to this guidance, one of which is that increased payload results in a decrease in the drone's flight time and range capacity [17]. Drones can support payload from 150 g to 830 g, depending on battery life [17]. Literature and research that have deployed drones for infrastructure inspection have indicated the use of commercial cameras with resolutions of 12 to 18 megapixel, with each 15-minute flight instance providing over 1200 images [18].

2.5 Space inspection robot systems

RAS systems have increasingly become a critical aspect of space technology, supporting a variety of space missions. Space robotics can be classified as either small or large manipulators, or humanoid robots [19]. Space robots, also referred to as space manipulators in some literature, meet decreased and applied gravitational forces on them. In most cases, these robots rotate and hover and glide in orbit [20]. Literature demonstrates two types of applications for space-based robotics, these include on-orbit assembly and on-orbit maintenance (OMM). OMM applications involve repair, refueling, debris removal, inspection, etc.

Since the scope of this paper is concerned with inspection purposed platforms, we review only those space robot systems that are developed to support inspection tasks. This includes the development of the orbiter boom sensor system (OBSS) by the Canadian Space Agency (CSA). The OBSS was deployed to inspect the façade of the thermal protection system of space shuttles [19]. Space robots can be deployed to ferry varied payloads of kilograms to tons on space installations [21].

[22] provided a description of the teleoperated robotic flying camera called the Autonomous Extra Vehicular Robotic Camera (AERCam), that is used to support astronauts by providing them with a way to inspect and monitor the shuttle and space station. The first version of the robot was called AERCam Sprint and was deployed on a shuttle in 1997. The AERCam Sprint was fitted with a ring of

twelve infrared detectors and two color cameras to enable vision capability [22]. [23] provided a report on a space inspection robot developed by NASA, called Tendril. The Tendril is a manipulator robot type purposed to support space missions by inspecting difficult to reach locations, e.g. fissures, craters, etc. Nishida *et al.* [24] provided a report on a prototype model for what they describe as an 'end-effector' for an inspection and diagnosis space robot. Pedersen *et al.* [25] mentioned a space robot designed to inspect the Mir station called Inspector, indicating however that it failed while in flight.

3 Features of data collected from RAS-based inspections

This section reviews the characteristics of data collected by RAS technologies, which are then processed and analyzed by analytical methods and techniques to support the monitoring and inspection of mechanical systems and civil infrastructure. We will explore the characteristics of input data collected by RAS systems using the four Vs data model [26–29]. Literature indicates that data can be categorized through its volume, which refers to the quantity of data that can be generated and stored; veracity, which refers to the quality of data collected as input; velocity, which refers to the speed at which data can be produced; and variety, which refers to the type and format of data collected. These four characteristics are briefly described below:

3.1 Volume (quantity or size of data)

Meyrowitz *et al.* [8] advised that there is a direct relationship between the type of sensor fitted to a robotic autonomous system and the volume of data produced. This position considers that certain sensors by default generate larger quantities of data compared to others, e.g., cameras which produce video data can generate millions of bits of data.

3.2 Variety (type and format of data)

Literature review showed that most research papers that have deployed climbing and crawler robots fit them with an array of sensors to collect a variety of data types. The data types include sound waves and their distance to an object, using ultrasonic sensors; acceleration and velocity using accelerometers and gravity sensors [13]. Climbing and crawling robots also collect image and video data using cameras [11, 13]. Literature review of the types of data collected by UAVs have demonstrated that they have been purposed to collect image and video data [15, 18, 30–32]. In a study by Alharam *et al.* [33], the UAVs have also been purposed to collect data on gas leakage, specifically methane (CH4) from oil and gas pipelines. The types of data collected by UUVs, specifically ROVs for underwater inspection, include image and video data; angular, velocity, orientation, depth, and pressure data, collected by optical and gyro sensors [7, 13]. The types of data collected by unmanned ground robots (UGRs) vary from images and videos collected by cameras; distance measurements collected by range finder sensors; and sound wave data collected by ultrasonic sensors [10].

3.3 Velocity (speed of data generation)

While the literature indicates that certain types of sensors produce higher volumes of data compared to others, it is also indicated that the speed of data generation and transmission has a direct correlation with the transmission medium or link used, and sometimes the environment the data is transferred within [8]. Except for onboard RAS data processing and analysis, data rates are slowed depending on their environment, Meyrowitz *et al.* [8] demonstrated that, with current technology, underwater RAS systems operate in an environment that reduce the rate of data transmission [8].

3.4 Veracity (quality and accuracy of data)

Literature indicates that the quality and accuracy of data can be directly linked to the frequency and type of transmission link used for data on RAS systems. In the instance of UUVs, Meyrowitz *et al.* [8] provided the instance of sonar imaging, where the combination of low frequency and bad transmission links result in reduced resolution and lots of interference that will require cleaning in image and acoustic data [8]. This has led to research into the development of better, high performance data transmission links, ranging from fibre optics to laser links. Alternatively, RAS systems with ML technology on-board, can process and analyze data with greater veracity because the data has not yet been subject to the degradation that is directly linked to transmitting the input data.

4 Machine learning techniques

A review of literature provides a distinct categorization between different ML algorithms. These algorithms are referred to as either supervised learning, unsupervised learning, or reinforcement learning algorithms. This section provides an overview of some of the popular ML techniques documented in literature and used to process and analyze data collected from RAS-based inspection operations.

4.1 Supervised learning

There are supervised learning techniques that are used to provide prediction-based solutions for problems that can either be categorized as classification or regression. These techniques require vast amounts of labelled data as input. In this approach to ML, the outputs (sometimes referred to as targets in literature) are pre-determined and directed towards interpretation and prediction. The dataset provided is separated into a training set and test set respectively, and then it is labelled with features of interest. When trained, the system can identify and apply these labels to new data, e.g., when taken through the supervised ML process, a system can be trained to identify new images of an object. Therefore, given input (x) and output (y), a supervised learning algorithm can learn the mapping function y = f(x); so that given input (x), this can lead to a prediction of output (y) for the given data [34–37].

There are two types of supervised learning approaches to ML. The first type is the regression learning process, where a model predicts a continuous quantity based on its input variables. Regression predicts continuous values such as height, weight etc., these values are referred to as continuous because they reside within an infinite number of possibilities, e.g., weight is a continuous value because there is an infinite number of possible values for a person's weight [38, 39]. The other type refers to the classification-based supervised learning process, where the output or target is categorical (discrete or finite number of distinct groups or classes). Classification refers to the process of identifying a model that takes a given dataset and sorts the data within it into distinct or separate classes or labels. Classification models in supervised machine learning are often described as a technique for predicting a class or label [34–37, 40, 41].

While most ML algorithms can be applied to solve both classification and regression problems, algorithms best suited to support classification-based problems include the K-nearest neighbors (KNN), logistic regression, support vector machine (SVM), decision tree, naive bayes, random forest, and artificial neural network (ANN). Conversely, algorithms best suited to support with regression-based problems include linear regression as well as random forest [39]. These techniques are briefly reviewed in the following.

4.1.1 Linear regression

Linear regression is a supervised model or algorithm used to predict the value of a variable based on the value of another variable. The model assumes that linear relationship between the input variables (x) and a single output (y), and that the output (y) can be calculated from a linear combination of the input variable (x). This is referred to as a simple linear regression. Literature describes multiple input variables as multiple linear regression. Linear regression models fit a straight line to a dataset, to describe the relationship between two variables.

4.1.2 Support vector machine (*SVM*)

SVM is a supervised machine learning algorithm that can be applied to both classification and regression approaches. SVM algorithm identifies a "decision boundary" or "hyper plane" to separate a dataset into two distinct classifications. The algorithm attempts to maximize the distance between the nearest data points of the two classes within the dataset. Support vectors are the data points nearest to the decision boundary; and a change in the position of the support vectors will result in the change in the position of the decision boundary. The greater the distance the data points are from the decision boundary, the more concrete their classification. The distance between the decision boundary and the nearest data point is called the margin. SVM is very accurate and works very well with small datasets. However, with large datasets it usually results in longer training times [33, 35, 42–44].

4.1.3 Decision trees (*DT*)

Decision Trees (DTs) are a supervised ML algorithm, that builds classification or regression models. DTs take the form of a tree diagram, breaking down a dataset into smaller subsets, to facilitate the development of a tree with a root node, a decision node, with each outward branch of the node representing a possible decision, outcome, or reaction. The decision tree comprises of decision nodes and leaf nodes, with leaf nodes representing a classification or a decision. DTs are typically used to determine a statistical probability or more simply, a course of action for complex problems. DTs provide a visual output of a given decision-making process and they can process both numerical and categorical data. However, DTs are susceptible to unbalanced datasets which generate biased models. DTs are also susceptible to overfitting - which occurs when a model fits too closely to the training data and is less accurate when introduced to new and previously unseen data [33–35, 42–44].

4.1.4 Random forest (*RF*)

Random forest is a supervised ML algorithm that can be applied to both classification and regression-based problems. It grows multiple individual decision trees for a given problem and merges them together to make a more accurate prediction. The RF technique uses randomness and ensemble learning to produce uncorrelated forests of decision trees. Ensemble learning is a method that combines various classifiers such as decision trees, and takes the aggregation of their predictions to provide solutions. The most commonly known ensemble methods are bagging and boosting. Bagging creates a different subset from the training data, with the final output based on majority voting. Boosting on the other hand is a method (e.g., ADA Boost, XG Boost) that combines "weak learners" into "strong learners" by creating sequential models so that the final model generated delivers the highest accuracy. Random forests, unlike DTs, are not susceptible to overfitting, however, they are a time-consuming and resource-intensive technique [33].

4.1.5 XGBoost (*Extreme Gradient Boosting*)

XGBoost is short for 'Extreme Gradient Boosting'. The XGBoost is a supervised ML algorithm implementing the gradient boosting decision tree framework. The algorithm can be applied to solve classification, regression, and prediction problems. It creates and works to optimize (through a boosting technique) each upcoming decision tree, so that the errors of each following decision tree are reduced compared to the previous tree that came before it. The boosting technique involves a process where there is a gradual learning from data, resulting in improved prediction for building subsequent decision iterations.

4.1.6 K-Nearest Neighbor (K-NN)

The KNN algorithm is a supervised ML algorithm, best suited to classification models. The algorithm makes an estimation of the probability that a new data point belongs to a particular group. This process involves looking at the data points in proximity and then identifying which data points have similar features to the new data point. The new data point is then assigned to the group which has most data points with similar features close the new data point. The KNN algorithm is very easy to implement and fast to execute. However, KNN does not classify data points very well and the accuracy of the algorithm is dependent on the quality of the dataset [35, 42–44].

4.1.7 Naive Bayes

Naive Bayes is a supervised ML technique used to solve classification problems, and is based around counting and conditional probability. It uses the Bayes theorem to classify data. The Naive Bayes algorithm naively assumes that all characteristics of a data point are independent of one another. The Bayes' theorem is based on the understanding that the probability of an event may require to be updated as new data becomes available. The algorithm seems to perform much better with categorical data (for example, it works well when applied to document classification and spam filtering) than with numerical data [34, 35, 42].

4.1.8 Logistic regression

Logistic regression is a supervised learning and classification algorithm for predicting a binary outcome, where an event occurs (True) or does not occur (False). The algorithm is used to distinguish between two distinct classes. It is considered a supervised ML algorithm because it has X input features and a y target value, and uses labels on the dataset for training. The algorithm works to find the logistic function of best fit to describe the relationship between X and y. Logistic regression algorithm is similar to the linear regression algorithm, except that the linear regression works with continuous target variables (numbers within a range) while the logistic regression is used when the target variable is categorical. The algorithm transforms its output using the sigmoid function to return a value which is then mapped to two or more discrete classes. Binary regression, multinominal logistic regression and ordinal regression are the three main types of logistic regression. Binary regression is used to process Boolean values, multinomial is used to process $n \geq 3$ values, and ordinal logistic regression processes $n \geq 3$ ordered classes. Logistic regression has been used to support various applications from medical diagnosis to fraud detection in banking [34, 42].

4.1.9 Artificial Neural Network (ANN)

Artificial Neural Networks (ANNs), which can solve both regression and classification problems, are modelled on the neural networks in the human brain. Like the human brain that contains billions of neuron cells that are connected and distribute signals in the human brain, ANNs are made up of artificial neurons, called units, grouped into three different layers. The first layer is called the 'input layer' which receives data and then forwards the data received to the second layer called the 'hidden layer'. The hidden layer performs mathematical computations on the data received from the input layer. The last layer is the output layer, which returns data as output. Deep neural networks (sometimes called deep learning in academic literature) refers to neural networks that contain multiple hidden layers [35, 42–46].

4.1.10 Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN) is a type of ANN that detects patterns and helps with processing of vision-based tasks. CNN is made up of an ML unit algorithm, called perceptions. A CNN can make predictions by analysing an image, check to identify features, and classify the images based on this analysis. CNN consists of multiple layers that process and extract features from data. These layers include the Convolutional Layer, Rectified Linear Unit (ReLU), Pooling Layer, and Fully Connected Network (FCN). The Convolutional Layer contains filters that perform the convolution operation, while the ReLU layer performs operations on elements and outputs a rectified feature map. The Pooling Layer takes the rectified feature map as input, and performs a 'down-sampling' operation that reduces the dimensions of the feature map. The pooling layer then converts the two-dimensional array output from the pooled feature map into a linear vector by flattening it. FCN layer takes the flattened matrix from the pooling layer as input and then proceeds to classify and identify the images [45–48].

4.2 Unsupervised learning

Literature tells us that unsupervised learning is where algorithms identify patterns within a given dataset. Unsupervised learning process involves searching for similarities that can be used to group data. Some of the most used unsupervised learning algorithms include the K-means clustering algorithm, Hierarchical clustering, Anomaly detection, principle component analysis (PCA), Independent Component Analysis, Apriori algorithm, singular value decomposition [34–37].

4.3 Reinforcement learning

Reinforcement learning (RL) takes an alternative approach to supervised and unsupervised learning. RL does not require the system to learn from data, instead learning is the result of feedback and reward. This involves a series of trial and error by a software agent. Some of the most common unsupervised learning algorithms include: SARSA – Lambda algorithm; Deep Q Network (DQN) algorithm; Deep Deterministic Policy Gradient (DDPG) and the Asynchronous Advantage Actor-Critic algorithm (A3C) [35–37].

4.4 Deep learning

The term deep learning refers to a subset of ML techniques that requires vast amounts of data to train models to output values, interpretations, or predictions. Deep learning methods are ANNs with more than one hidden layer and they can be supervised or unsupervised. Applications that require the application of deep learning techniques as supervised learning, include image classification, object detection and face recognition. Alternatively, applications that require the deep learning techniques as unsupervised learning are usually instances where there is no labelled data and for clustering problems, e.g., image encoding and word embedding.

4.4.1 Deep Neural Network (DNN)

A deep neural network (DNN) are ANNs that have more than one hidden layer (therefore the term "deep"), that are trained with vast amounts of data. Each hidden layer comprises of neurons that map a function to input to provide an output. DNNs are trained through the adjustment of its neurons, biases and weight features. These types of neural networks are also supported by various techniques such as the back-propagation algorithm and optimization methods such as stochastic gradient descent. Three types of deep neural networks include multi-layer perceptrons (MLP), convolutional neural networks (CNN) and recurrent neural networks (RNN). DNN features support speech recognition systems, and translation systems like Google Translate [49, 50].

4.4.2 Deep Belief Networks (DBNs)

Deep belief networks (DBNs) consist of unsupervised networks, that comprise of a stack and sequence of connected restricted Boltzmann machines (RBMs). The DBN trains each of the Boltzmann machine layers until they converge. The value of the output layer of a Boltzmann machine is input into the next Boltzmann machine in the sequence, then again trained until convergence is reached. This process is repeated with each Boltzmann machine until the whole network has been successfully trained. Applications of DBNs vary from generating of images to video sequences and motion capture [51–53].

4.4.3 Recurrent Convolutional Neural Networks (RCNN)

Recurrent Convolutional Neural Networks (RCNN) algorithms detect and localize objects in an image. This is done by drawing rectangular boundary like boxes around objects contained within an image, placing a label on, or categorizing each defined box in an image, extracting features in the image using the SVM algorithm, and then processing the features using a pre-trained CNN. The last stage in the process brings separate regions together to obtain the original image with the identification of the objects within the image [47, 54–58].

4.4.4 Fast R-CNN

An iteration or evolution and improvement of the R-CNN model can be found with the Fast R-CNN algorithm. Fast R-CNN model takes the image as a whole and passes it to its neural network to output, the output is then sliced into region of interests (ROI).

4.4.5 Faster R-CNN

A further evolution of the R-CNN model is the Faster R-CNN algorithm [59]. Faster R-CNN is a better performing and faster algorithm than R-CNN and Fast R-CNN, because it only uses CNNs and does not use SVMs, and provides a single feature extraction of an image, instead of region-by-region extractions of an image like R-CNN. According to literature, this results in Faster R-CNN training networks at least nine times faster, with more accuracy than R-CNN [59, 60]. However, what makes Faster R-CNN distinct to its predecessor Fast R-CNN is the use of the Region Proposal Network (RPN) technique [60].

4.4.6 Mask R-CNN

The Mask R-CNN is an extension of the Faster R-CNN technique. Literature describes the Mask R-CNN technique as an advanced image segmentation method, which takes a digital image and breaks it down into segments or pixels, and then categorizes the segments. For example, a single image is segmented and categorized to identify multiple objects in the image [61].

4.4.7 R-FCN

Literature describes the R-FCN model as being based on region proposal. The difference between the R-FCN and R-CNN techniques (which is also based on region proposal) is that R-FCN applies the selective pooling technique that extracts features for prediction on the last layer of its network [62].

4.4.8 Single Shot Detector (SSD)

The Single Shot Detector (SSD) is an ML technique that breaks down an image into a grid of cells. In turn each cell has the function of detecting objects, by predicting the category and location of objects in the region where the images are located within. Literature indicates that the SSD model is faster than the Faster R-CNN model. However, when the object size is small, the model's performance decreases [63].

4.4.9 You Only Look Once (YOLO)

The YOLO (You Only Look Once) algorithm uses CNNs to detect and recognize objects in a picture in real-time. YOLO first takes an entire image as input, divides the image into grids (this is different to R-CNN that uses regions to localize objects in an image), image classification and localization are applied to each grid. The algorithm then predicts the rectangular (bounding) boxes and their associated classes. The YOLO model does however find it difficult to localize objects properly compared with R-CNN [37, 64].

4.4.10 Recurrent Neural Networks (RNNs)

Classic neural networks are described as 'feed forward' networks because they channel information in a single forward direction, through a series of mathematical operations performed at the nodes of the network. Data is fed through each node as input, never visiting a node more than once, before being processed and converted into an output. Feed forward networks only perceive the current sample data that has been provided in present time and have no facility for memory with respect to previous data samples processed. In other words, classic neural networks do not have the facility for data persistence.

RNNs are a type of deep neural network, and unlike classic neural networks they take both the current data sample and the previously received samples as input. RNNs can process data from the first input to the last output and initiate feedback loops throughout the entire computation process, enabling the loop of data back into the network. RNNs are distinct from feed forward networks by the feedback loop connected to their past decisions. RNNs allow previous outputs to be provided as inputs, while also having hidden states. RNN models are commonly used in the natural language processing (NLP) and speech recognition domain [65, 66].

4.4.11 Long Short-Term Memory Networks (LSTMs)

LSTMs are a special type of RNN that help preserve the error that can be back propagated through layers and time. LSTMs provide recurrent networks with the ability to learn over time. This is made possible in large part to LSTMs' gated cell, from which data can be written to, read from and stored into, all external to the back and forth of the recurrent network [67].

4.4.12 Generative Adversarial Networks (GANs)

GANs are described as generative deep learning unsupervised learning algorithms. The technique was introduced in 2014 by Ian Goodfellow. The premise of GANs involves a neural called a generator, which produces fake data samples. The generator works in concert with another network called the discriminator which has to differentiate between two different input data samples. The first being the original data samples and the second being the fake data samples being created and output by the generator. The discriminator has to evaluate, learn and make decisions as to which data sample is from the actual training set and which are form the generator [68, 69].

4.4.13 Multilayer Perceptrons (MLPs)

A perceptron is an input layer and an output layer that are fully connected; and comprise of input values, weights and bias, net sum, and an activation function. A fully connected neural network with multiple layers is called Multilayer Perceptron (MLP). MLP is a supervised learning feed forward deep neural network that connects multiple layers in a directed graph, in other words, where the signal path is through a single direction through the nodes, between the input and output layers. In this network, every node, with the exception of the input nodes, contains a non-linear activation function. MLPs can be used to build speech-recognition, image-recognition, and machine-translation applications [70, 71].

4.4.14 Restricted Boltzmann Machines (RBMs)

Boltzmann machines are non-deterministic, generative deep learning models with only two types of nodes - hidden and visible nodes. There are no output nodes, which provides them with the non-deterministic feature. Boltzmann Machine has connections among the input nodes, with all nodes connected to all other nodes including input or hidden nodes. This allows universal information sharing of parameters, patterns and correlations of the data. Restricted Boltzmann Machine (RBM) are a special class of Boltzmann Machines. RBM is an unsupervised two-layered (visible layer and hidden layer) neural network. RBM is characterized by restrictions where every node in the visible layer is connected to every node in the hidden layer but no two nodes in the same group are connected to each other [51–53].

4.4.15 Autoencoders

Autoencoders are an unsupervised type of neural network that can be used to detect patterns or structure within data to learn a compressed representation of data provided as input. The autoencoder learns how to compress the data based on its attributes during training. An autoencoder is a feed forward neural network where the input is the same as the output. It is made up of encoder and decoder models. The encoder compresses the input, and the decoder works to re-create the input from the compressed version of the input that was provided by the encoder. Applications of autoencoders vary from anomaly detection, data denoising (audio and images), dimensionality reduction etc. [72–76].

Table 1 reviews the advantages and disadvantages of different ML techniques reviewed in Section 4 of this paper.

Table 1 Review of advantages and disadvantages of different Machine Learning techniques

Technique	Design	Advantages	Disadvantages
SVM (Support Vector Machine)	An SVM algorithm, through the use of its decision boundary, attempts to maximize the distance between the nearest data points of the two classes withn the dataset [33, 35, 42–44].	SVM algorithms work well on small datasets. The algorithm also provides greater accuracy than contemporary algorithms.	SVM algorithms do not work well with large datasets. Training times with large datasets are high.
Naive Bayes	The Naive Bayes algorithm naively assumes that all characteristics of a data point are independent of one another [34, 35, 42].	The Naives Bayes algorithm is easy to implement, can be trained with ease, and scales well with a given dataset.	Performance is significantly reduced when independent assumption does not hold. Method provides zero probabilities with discovery of words in a class in test data, not found in training data.
Logistic Regression	Logistic regression is a classification algorithm for predicting a binary outcome, where an event occurs (true) or does not occur (false) [34, 42].	Logistic regression is simple to implement. It is an optimal algorithm for analysis and working with binary data.	Logistic regression algorithms perform poorly on non-linear datasets e.g., images. The algorithm also performs quite poorly next to other algorithms.
Decision Trees	DTs take the form of tree a diagram, the algorithm breaks down a dataset into smaller subsets. The decision tree comprises of decision nodes and leaf nodes, with leaf nodes representing a classification or decision. DTs are typically used to determine a course of action for complex problems [33–35, 42–44].	They provide a user-friendly interpretable and visual method for solving problems and making decisions. DTs can manage numerical and categorical data. They also require minimal data preparation.	Results from decision trees can be unstable. DTs are sensitive to the specific features in a dataset and therefore small changes to the data result in significant changes to a tree, sometimes providing a completely different tree. This problem has been mitigated with the application of techniques such as 'Bagging' and 'Boosting'. Boosting is where (in the case of regression trees) much new, stronger trees (trees with less errors) grown, are spawned from a collection of weaker trees (trees with more errors) [77].
Random Forest	Random forest is a supervised machine learning algorithm that grows multiple individual decision trees for a given problem, and merges them together for a more accurate prediction [33].	The random forest algorithm can manage large datasets quite efficiently, with more accurate predictions. The algorithm also works well with datasets with missing data.	The algorithm can be time consuming when handling large datasets, because of the computing time required to build, manage and merge all the individual trees. The algorithm requires and consumes more resources, especially when managing large data sets, requiring storage. The algorithm is more complex as the prediction of an individual decision tree is easier to interpret compared to a forest of trees.
Linear Regression	Linear regression is a model or algorithm used to predict the value of a variable based on the value of another variable. Linear regression models fit a straight line to a dataset, in order to describe the relationship between two variables.	The model is easy to interpret and implement.	The linear regression model oversimplifies real-world problems by assuming a linear relationship (or straight line) between variables. The model is also very sensitive to outliers (anomalies) and therefore requires the extraction of outliers from a dataset, before the model can be applied.
XGBoost (Extreme Gradient Boosting)	The XGBoost algorithm creates and works to optimize (through a boosting technique) each upcoming decision tree, so that the errors of each following decision tree is reduced compared to the previous tree that came before it.	The XGBoost algorithm performs very well on small datasets and handles mission data well. The algorithm provides a better prediction accuracy compared to other machine learning algorithms.	Technique is sensitive outliers and does not perform well on unstructured data.
K-NN (K Nearest Neighbors)	The K-nearest-neighbor makes a guess or estimation of the probability a new data point belongs to a particular group [35, 42–44].	The KNN algorithm is simple to understand and implement. The algorithm adapts with new data and does not not make assumptions about data.	The prediction process runs quite slowly when working with large datasets, and the algorithm does not work well with imbalanced data. The algorithm is sensitive to outliers and cannot manage missing values. KNN also requires a lot memory to store the training data, which also makes it a very computationally expensive algorithm.

Table 1 (*Continued*)

Technique	Design	Advantages	Disadvantages
ANN (Artificial Neural Networks)	Artificial neural networks (ANN) are made up of artificial neurons, called units, arranged in a series of layers [35, 42–46].	ANN algorithms can be applied to both classification and regression problems. Neural networks can model non-linear data with a large number of inputs e.g., images, audio and text.	ANNs require a lot of data to train, and are computationally expensive. Neural network algorithms run slow with traditional CPUs and require GPUs.
CNN	A convolutional neural network (CNN) can make predictions by looking at an image and checking to identify features, and classifying the image based on this analysis [30, 31, 45–48].	CNN algorithms independently and automatically detect features, and are computationally efficient.	CNN model cannot detect objects within an image, in other words they cannot tell us where the objects are located. CNNs do not encode the position and orientation of an object in their data routing between layers and in the resulting output or predictions [47].
R-CNN	The R-CNN algorithm takes an image, extracts regions (or features) in the image using a selective search method. The region of interest (ROI) is placed in a rectangle that represents the boundary of an object in the image. R-CNN then processes these features using a pre-trained CNN. It then classifies the extracted regions using a support vector machine (SVM), which involves determining what type of object is in the ROI boundary. Finally combining the regions to obtain the original image with objects detected in it [47, 54–58].	The R-CNN model can both detect objects in images as well as classify them.	Multi-step process requires a lot of time to train the network because of the significant number of regions (or features) it needs to classify for every image. Model also requires a lot of disk space and is computationally expensive algorithm. Bounding box generated does not give any indication about the shape of the object.
Fast R-CNN	In Fast R-CNN, the whole image is passed to a neural network to produce output features. Then, a new technique called ROI Pooling slices out each ROI from the network's output sensor. Similarly, to R-CNN, Fast R-CNN also uses the selective search method to generate its region of interests.	Fast R-CNN is faster than R-CNN because it passes the whole image to neural network to produce the output features, unlike R-CNN which processes each ROI individually.	Technique is still slow for real world.
Faster R-CNN	Faster R-CNN is a better performing and faster algorithm than R-CNN because it only uses CNNs and does not use the selective search method or SVMS, and provides a single feature extraction of an image, instead of region by region extractions of an image like R-CNN. According to literature, this results in Faster R-CNN training networks at least nine times faster, with more accuracy than R-CNN [59, 60]. However, what makes Faster R-CNN distinct to its predecessor Fast R-CNN, is the use of the Region Proposal Network (RPN) technique [60].	Faster R-CNN builds a single stage network, which is more efficient and better performing than R-CNN's multiple stage process. Faster R-CNN also uses the ROI pooling technique to share computation data for the processing of all objects in an image.	The process of sampling of multiple regions from an image results in performance degradation.
Mask R-CNN	Mask R-CNN provides instance segmentation as well object detection. Mask R-CNN substitutes an ROI Pooling method with an ROI Align method, which can represent fractions of a pixel.	Simple to train and provides good performance compared to other models.	Reduced ability to predict instance details.
R-FCN	R-FCN stands for Region-based Fully Convolutional Network.	R-FCN is faster than Fast R-CNN or Faster R-CNN.	Cannot be implemented in real-time and still takes a long time to train.
Recurrent Neural Networks (RNNs)	RNNs are a type of deep neural network. RNNs can process data from the first input to the last output, however, RNNs are also able to initiate feedback loops throughout the entire computational process, enabling the loop of data back into the network.	Advantages include the facility to process input of any length, the network computation considers historical information.	Disadvantages with RNNs include the fact that computation can be slow, and the network does not consider future input for the current state.

Table 1 (*Continued*)

Technique	Design	Advantages	Disadvantages
Long Short-Term Memory Networks (LSTMs)	LSTMs are a special type of RNN that help preserve the error that can be back propagated through layers and time. LSTMs provide recurrent networks with the ability to learn over time. This is made possible in large part to LSTMs' gated cell, from which data can be written to, read from and stored into, all external to the back and forth of the recurrent network [67].	LSTMs were developed to manage the vanishing gradient problem when training RNNs. LSTMs provide us with a large range of parameters such as learning rates, and input and output biases.	LSTMs require a lot of resources and time to get trained and become ready for real-world applications. They are also affected by different random weight initialization and prone to overfitting.
Generative Adversarial Networks (GANs)	The premise of GANs involves a neural called a generator, responsible for producing fake data samples. The discriminator network has to evaluate, learn and make decisions as to which data sample is from the actual training set and which are from the generator [68].	GANs are unsupervised, therefore no labelled data is required to train them, this is cost-saving since data labelling is an expensive task. GANs currently generate the sharpest images.	GANs are more difficult to train, you need to constantly provide different types of data to check if it works and is accurate. Furthermore, given that the generator and the discriminator networks compete against each other during the training, there is always the possibility that the discriminator network learns too quickly, becomes successful, the other network may fail to learn and the generator gradient disappears and learns nothing. This can often result in the networks failing to converge.
Multilayer Perceptrons (MLPs)	MLP is a supervised learning feed forward deep neural network that connects multiple layers in a directed graph, in other words, where the signal path is through a single direction through the nodes, between the input and output layers. In this network, every node, with the exception of the input nodes, contain a non-linear activation function [70, 71].	MLP networks work well with large input data and can provide fast and timely predictions after training.	The nodes in an MLP are all tightly connected to one another, sometimes resulting in unwanted redundancy. Computations can be difficult and time consuming.
Deep Belief Network (DBN)	Deep Belief Network (DBN) is deep neural networks, where each layer learns about the input one at a time and uses unsupervised learning to output data. DBN uses a greedy learning algorithm to train its networks, where the selection of an optimal decision is made at each layer in the network, to result in a global optimum decision.	DBNs require a small labeled dataset; also need a short training time on GPU machines; and less vulnerable to the gradient problem.	DBNs place significant of restrictions on weight connections, and deep neural networks (DNNs) outperform them on tasks for which sufficient input data is available.

4.5 Performance evaluation metrics for machine learning techniques

In this section, we examine the academic conventions in literature for measuring the performance of machine learning techniques in the detection of material defects and equipment failures. Current academic literature has provided methods for evaluating the performance of computer vision and especially machine learning methods and techniques, when applied to extract, process and analyze datasets. The literature indicates that while ML techniques can be used to extract and analyze data, these techniques however can also generate false results due to misclassification or misinterpretation of data collected. This is the reason for performance measurement metrics, which evaluate the performance of ML methods and the ratio of correct predictions or classifications to misclassification or incorrect predictions.

Classifiers or classification-based ML techniques in literature usually use the confusion matrix, accuracy/error, precision, recall, F1 measure (or F-measure), ROC, AUC, Hypothesis test (t-test, Wilcoxon signed-rank test, Kappa test) as the metric system or evaluation measurements [78]. Regression based problems use MSE (mean squared error), MAE (mean absolute error), MAPE (mean absolute percentage error), RMSE (root mean squared error) which is the square root of the average distance between the actual score and the predicted score, and quantiles error for evaluating the performance of machine learning methods applied as solutions [79].

4.5.1 Confusion matrix

Given that classification is when the output data is one or more discrete labels, regression however is when the output data or prediction of the model is a continuous quantity or value. Binary classification is where there are only two class categories of vector y (target label) (e.g., True or False, 1 or 0, etc.) in the dataset. Conversely, multi-class classification is when there are three or more classes or categories of the vector y (target label) in the dataset. The performance measure of ML techniques, specifically for classification tasks (e.g., binary classification, multi-class classification) are commonly analyzed using a confusion matrix, which is a two-dimensional matrix that measures the False Negative (FN), True Negative (TN), False Positive (FP), and True Positive (TP) for each model used. TP is in reference to the positive points that are correctly labelled by the classifier. TN refers to the negative points that were correctly labelled by the classifier. FP are the negative points that were incorrectly labelled as positive, and FN are the positive points that we mislabeled as negative.

These measurements are then used to calculate the performance metrics: recall, F1 score, accuracy, and precision. The accuracy is a percentage of predictions that are correct (TP + TN). The precision measures how accurate the model being used is for predicting values. The recall measures the sensitivity of the model to predict positive outcomes. The F-measure is a combination of the precision score and recall score [33, 80].

4.5.2 Mean Square Error (MSE) and Mean Absolute Error (MSD)

The performance measure of a regression model is commonly analyzed using either the mean square error (MSE) or the mean absolute error (MSD). The MSE measures the average of the squares of the errors, that is the average squared difference between the predicted value and the target value. The lower the MSE, the better. The MSD measures the average of the absolute differences between model prediction and target value.

4.5.3 Mean Average Precision (MAP)

Some studies have recommended the use of the evaluation metric called the mean average precision (MAP) to analyse the performance measure of object detection (localization and classification) models such as SSD, R-CNN and Faster RCNN and YOLO [31]. The MAP is also commonly applied to analyze the performance of computer vision models and image segmentation problems. In MAP, an object (the predicted value) is taken as accurate on the condition it overlaps with what is labelled as the ground truth (the original damage annotated by human inspectors) that is greater than a given threshold. This is calculated using the Intersection over Union measure, which is expressed as the area overlap (predicted value overlap with the target value) over the area of union (area of the union of the predicted value and target value) [81].

5 Findings of literature review

This section provides an overview of the findings of the literature review in terms of types of mechanical systems and civil infrastructure assets. We then proceed to examine different types of damage mechanisms that can be found on these assets. This is followed by a review of our findings regarding the application of ML techniques used to support RAS-based monitoring and inspection.

5.1 Types of assets under inspection

We begin the review of our findings with discussing the types of various mechanical systems and civil infrastructure assets that literature indicate are subject to routine inspection and monitoring due to their vulnerability to catastrophic damage.

5.1.1 Pipelines

Pipelines in the energy industry support the transport and distribution of water, oil, and gas. Similar to most infrastructure, pipelines are subject to internal and external mechanical stresses. Such stresses can lead to damage mechanisms ranging from corrosion, cracks scale formation etc. This phenomenon can be mitigated through regular monitoring, inspection, and maintenance. There has been some research aimed at methods and techniques to support the inspection of pipes or pipelines more efficiently, this research ranges from Mohamed *et al.* [82], who looked at the use of mobile in-pipe inspection robots (IPIR) for inspection of corrosion and cracks in pipelines, using NDT sensors, e.g., ultrasonic. There is also the work carried out by Bastian *et al.* [30], who applied CNN architecture-based techniques to detect corrosion from pipeline images and DNN to extract features from those images.

5.1.2 Wind turbines

Wind turbines are another part of the energy industry and are generally located in remote environments, which are subject to extreme external stresses, e.g., wind, water, heat, etc. Similarly, just like pipelines, wind turbine infrastructure is also subject to mechanical stresses resulting in damage mechanisms such as erosion, cracks, etc. This is the reason for the emphasis and importance placed on regular inspection, monitoring and maintenance in industry planning and cost allocation. While wind turbine inspection is generally carried out using traditional methods, involving manual climbing of heights by technical inspection teams, in sometimes hazardous conditions, the last decade has seen a lot of research into the development of cost effective and safe methods and techniques that support the inspection and monitoring of asset infrastructure. This research has provided the alternative use of robot platforms, which vary from magnetic climbing robots to unmanned aerial drones. Current research has also seen the development of artificial intelligence-based algorithms and techniques that will analyze the vast amount of data collected by robot platforms, during inspection. Wang and Zhang [2] has researched the use of Haar-like feature algorithms for the automatic detection of surface crack images collected by UAVs; while Frank *et al.* [83] proposed applying a deep convolutional neural network (DCNN) technique on images from a wind turbine, taken by a multi-robot system. Shihavuddin *et al.* [31] carried out research using convolutional neural network (CNN) techniques to extract feature descriptors from images of wind turbines taken by drones; and the Faster R-CNN technique to train for object detection.

5.1.3 *Aircraft fuselage*

In the aerospace industry, the aircraft fuselage is one of the core and most important regular inspection tasks performed by maintenance technicians. The process involves deploying platforms that elevate technicians so that they can reach and inspect the external surface of the aircraft's fuselage, searching for damage mechanisms or defects. The main damage or defect mechanism found on aircraft surfaces is corrosion. Corrosion is a main cause of fuselage fatigue. Alongside checks for corrosion, technicians also look for rust, cracks, and deformation of the aircraft surface during their inspections. This process has traditionally involved a painstaking, methodical task, undertaken by a technician, equipped with a flashlight and mobile elevation platform [84]. Research into alternative methods of inspection have been published over recent years. This includes Malekzadeh *et al.* [85] work which applied deep learning techniques: SURF, AlexNet and VGG-F to images of aircraft fuselage, taken by a custom-made platform. Miranda *et al.* [86] did similar work when they applied a CNN based application, along with an SVM model to images of an aircraft fuselage, taken by UAVs. The most recent work is research carried out by Brandoli et al. [84] which involved applying CNN models: DenseNet and SqueezeNet to detect corrosion pillowing in images taken from an aircraft fuselage.

5.1.4 *Power lines*

In the energy industry, power transmission lines act as connections between the source of power (the power plants) and endpoints (the consumers) [87]. The regular inspection of power transmission lines is considered vital in ensuring uninterrupted power supply, as damage to this part of the electrical infrastructure through rusted conductors etc. can result in downtime and power interruption [88]. As indicated by Titov et al. [88], the traditional operation for power line inspection involves a process characterized by high cost and safety risk, where human technicians manually taking images from the ground or from above with the support of a helicopter. There has been research in recent years that have demonstrated the use of UAVs (drones) to support the collection of image data. Jalil *et al.* [87] carried out work on power lines, employing a drone, fitted with image capture equipment to collect data, which is then passed on to a neural network model for detection and analysis of damage or defects from the image dataset. Research undertaken by Titov et al. [88] employed UAVs to take images of power lines for detection and analysis of cracks on concrete poles, missing or dirty insulator plates etc. using the YOLO version 3 deep learning technique.

5.1.5 *Vessels*

The maintenance of vessels, especially maritime transport ships such as oil tankers or very large crude carriers (VLCC) require regular monitoring and inspection schedules [89, 90]. These vessels are subject to typical internal and external phenomenon such as cracks, corrosion etc. Current inspection procedures for vessels today require are expensive and the vessel to dock at a shipyard, where inspectors, with the support of all sorts of mobile platforms undertake visual assessment of the structural health and condition of the vessel. Recent research over the years have studied the use of robot platforms in support of vessel inspections [89, 90]. This research has been accompanied with the development of various image processing and damage (e.g., corrosion and cracks) detection algorithms to analyze image datasets. However, there is evidence of very little to no research of the use of deep learning (neural network) based techniques.

5.1.6 *Bridges*

Bridges are typical civil infrastructures that are subject to external phenomenon such as wind, heat, water, and vibrations. The inspection of bridges currently requires manual visual inspection by human inspectors, working at different levels of elevation, with various levels of risk associated, to access and view parts of the bridge. This manual approach requires high time durations, in some cases road closures, high costs and due to the vast number of bridges in cities today, a lot of manpower. The last decade has seen quite a significant amount of research into the use of robot and autonomous systems (RAS) to support bridge inspections. These robot platforms are fitted with sensors ranging from infrared (IR) cameras to ultrasonic sensors [14, 91]. This has been accompanied with research into computer vision and image processing techniques that detect damage mechanisms, defects, and features (e.g., cracks) [1]. This research has recently moved towards the use of deep learning methods to analyze images and detect cracks [49, 92, 93].

5.1.7 *Automotive vehicles* (*cars*)

Currently most literature available has concentrated on the analysis of image datasets have been focused on vehicle make and model classification in support of the transport and security industry [94] defect and damage detection and analysis in support of the automotive industry and the insurance sector that supports it [95, 96]. Most automotive inspection required are focused on analysis of vehicular accidents, requiring image analysis of damages such as bumper dents, door dents, glass shutters, broken head lamps and tail lamps, as well as scratches etc. [95–97]. Recent years has produced research in the development of computer vision and neural network techniques for image analysis of car damage [95–99].

5.2 Damage mechanisms on mechanical systems and civil infrastructure

Corrosion and cracks are damage mechanisms associated with bridges, roads, rail, and levees in literature [1, 49, 93, 100]. Most of the current literature focus on the inspection of wind turbines on surface damage, specifically, leading edge erosion, surface cracks, damaged lightning receptors and damaged vortex receptors [2, 31, 83]. In the case of pipelines, literature focus inspection of cracks, corrosion, or erosion [30, 33, 37]. Literature discussions of aircraft inspection consistently focuses on the fuselage, for damage mechanisms such as corrosion pillowing and insulator surface cracks. Some literature on aircraft inspection however do not specify damage mechanisms, and instead uniformly refer to defect regions [84–86].

Power lines and transmission lines in literature, are associated with cracks on concrete poles, identification of missing or dirty insulator plates, rusted conductors, broken cables, insulator damage, conductor corrosion and cracks on insulator surfaces [87, 88, 101, 102]. The concentration on the inspection of vessels or ships in literature surrounds the detection, localization and classification of cracks, corrosion, or coating breakdown, pitting, and buckling [89, 90, 103].

While there could be a case for the monitoring and inspection of automotive vehicles during servicing and checkups for damage mechanisms from internal and external phenomenon (e.g., wind, water, heat), such as corrosion, fatigue etc. there seems to be little to no literature on such research available at this time. The literature available is mostly focused on damage to cars because of vehicular accidents [95–99]. A review of literature shows that corrosion and cracks are the most addressed damage mechanism, while erosions and fatigue seem to be the least addressed damage mechanisms. Table 2 demonstrates the typical research into damage mechanisms on mechanical infrastructure, that machine learning techniques have been used to detect, classify, and model.

Based on the literature reviewed, manual visual, data collection is still the norm for bridge inspection, however, in research where a robot platform is deployed to collect image datasets of damage mechanisms on bridges, most papers have reported the deployment of UAVs [49, 93, 100]. Likewise, literature on wind turbine damage inspection showed a majority preference for drones to support with data collection [17, 31, 83].

In the case of pipelines, the literature shows a mixed picture, with a balanced preference for deploying drones for some, and a preference for the use of mobile in-pipe inspection (IPIR) robots for others [33, 37]. The literature, however, provides a different landscape for the use of RAS systems in aircraft inspections. While some literature has demonstrated the use of UAVs in recent years, the literature however, indicates a preference for the use of D-Sight Aircraft inspection system (DAIS) platforms. These are portable non-destructive devices, that support the visual analysis of aircraft surface areas of the fuselage [84–86].

While in some cases, piloted helicopters are still used to gather data, however, literature demonstrates that power lines and transmission lines inspection overwhelmingly deploy UAVs for data collection of damage mechanisms [87, 88, 101, 102]. Literature regarding vessels or ships, present a very mixed preference for the use of semi-autonomous micro-aerial vehicles (MAVs) and more recently, climber or UAV robot platforms for inspection and data collection of damage mechanisms [89, 90, 103]. Literature review of automotive vehicles for cars, has unfortunately provided very little to no research with respect to the use of robot and autonomous system platforms for damage inspection.

5.3 Application of machine learning techniques for RAS inspection

This section reviews the application of machine learning techniques used to support RAS system inspection of civil and mechanical infrastructure (wind turbines, pipelines, rail, aircraft fuselage, power lines, vessels, and automobiles) in literature and performance evaluations of their use where documented. Gopalakrishnan *et al.* [49] looked at the processing and analysis of images of cracks on civil and mechanical infrastructure, obtained through UAVs (or drones). They applied deep convolutional neural networks (DCNN) models to process and analyze the image data and lauded the efficacy of DCNN as the more efficient technique for processing and analyzing both images and video type data [49]. Nguyen *et al.* [104] reviewed the use of various deep learning techniques in support of RAS systems inspection of power line infrastructure and indicated that Region-Based Convolutional Neural Networks (R-CNN) and the You Only Look Twice (YOLO) techniques as the optimal techniques for object detection for inspection tasks by RAS systems.

Shihavuddin *et al.* [31] explored the use of drones to obtain image data of damage on wind turbines (WT), for processing, analysis and classification using deep learning techniques. The research paper indicates that the research used Convolutional Neural Networks (CNN) as the

Table 2 Review of collection and analysis of damage mechanisms analysis on civil and mechanical infrastructure

Asset Infrastructure	Industry	Damage (or Defect) Mechanism	Robotic Autonomous System (RAS)	Technique	References
Bridges, roads, rail and levees etc.	Transport	Cracks/Corrosion/	UAV (Drones)/Autonomous mobile robotic system	Deep Convolutional Neural Network (DCNN)/Navigation algorithm; Motion Planning Algorithm; Crack Detection Algorithm/Mask R-CNN/modified deep inception neural network (DINN)	[1, 49, 93, 100]
Wind turbines	Energy	Surface damage: leading edge erosion; surface cracks; damaged lightning receptors; damage vortex generators etc.	UAV (Drones)	Convolutional Neural Network [CNN: for extracting feature descriptors]/Faster R-CNN: to train for object detection/Haar-like features/Deep Convolutional Neural Network (DCNN)	[2, 31, 83]
Pipelines	Energy	cracks, corrosion or erosion (resulting in gas leakage)	UAVs/Drones (multi-rotor)/mobile robot (IPIR - In-Pipe Inspection Robot)/	Decision Tree (DT)/Support Vector Machine (SVM)/Random Forest (RF)/CNN (architecture to detect corrosion in pipeline images)/DNN (for feature extraction)/	[30, 33, 37]
Aircraft fuselage	Aerospace	Corrosion pillowing/Damage mechanisms not specified, paper refers to damage mechanisms as defects regions only/Defects e.g. cracks on insulator surface/	D-Sight Aircraft inspection System (DAIS) - portable non-destructive device for visual analysis of aircraft surface areas aimed at inspecting fuselage joints used/UAVs	DenseNet/SqueezeNet/SURF/ AlexNet/VGG-F/SVM/CNNs/	[84–86]
Power lines (or insulators)	Energy	Defects, e.g., cracks on concrete poles, missing or dirty insulator plates etc./rusted conductors, broken cables, insulator damage, conductor corrosion/Defects e.g. cracks on insulator surface/	UAVs (or piloted helicopter)	YOLO v3/CNN/DCNN/SVM	[87, 88, 102]
Transmission lines	Energy	Insulator defects	UAVs	[i] CNN [II] DCNN	[101]
Vessel	Transport	coating breakdown or corrosion, cracks, pitting, cracks, buckling	semi-autonomous multi-rotor micro-aerial vehicles (MAVs)/UAVs, Wall Climbers/	Feed forward neural network (FFNN)/image processing algorithms/Crack and corrosion detection algorithm/	[89, 90, 103]
Automotive Vehicles (Cars)	Automotive	bumper dent, door dent, glass shatter, broken head lamp, broken tail lamp, scratches, smashes	Not Applicable	CNN (ImageNet/AlexNet)/Mask RCNN/Custom Mask RCNN/	[95–99]

backbone framework for processing and extracting features from image data obtained from WT using drones. Shihavuddin report their research proceeded to then use the faster R-CNN technique to train their models for object detection. They report achieving high accuracy results compared to other deep learning type algorithms used to train the models during their research. Shihavuddin *et al.* [31] also indicated that employing the technique called advanced image augmentation, allows you to expand your dataset, as this technique creates additional images for the training model, by altering existing image data obtained and fed into training sets. The utility of this technique is invaluable, as the larger the dataset, the more efficient the training model.

Franko *et al.* [83]'s research provieded findings in the use of a combined, multiple RAS platform, ranging from climbing and multicopter robots, fitted with LiDAR, RGB and ZED cameras, ultrasonic, radar and other vision type sensors, to inspect and detect corrosion and welding line damage mechanisms on the tower surfaces of WTs [83]. Alharam *et al.* [33] provided a case study in the use of UAVs to provide inspection of oil and gas pipelines in Bahrain. The UAVs are fitted with GPS, thermal cameras, and gas detectors to obtain image and methane (CH4) readings from gas and oil pipelines. The research looked at the use of the Decision Tree (DT), Support Vector Machine (SVM) and Random Forest (RF) techniques to process and analyze the data obtained from the drones. Franko *et al.* [83] reported that the RF technique provided 93% accuracy and much better performance than other classification techniques used in their research [33].

Bastian *et al.* [30] studied the external corrosion on pipelines and used deep neural networks to process and analyze the image and video data obtained from inspections. In their research paper, they proposed the use of a DNN technique, based on the CNN architecture, to extract and distinguish between images with corrosion and those without, from image data taken from pipelines by the UAV. They boast in their paper that CNNs have the most optimal results in terms of object detection and image classification.

Table 3 shows us that the literature demonstrates that the DCNN architecture provides up to 90% accuracy for the detection of cracks in civil infrastructure [49]; 92% accuracy in defect detection in rail infrastructure [105]. Table 1 also indicates that while the Random Forest algorithm is the best performing algorithm when compared to the Decision Tree and Support Vector Machine algorithms for detection of cracks, corrosion, and erosion on pipelines, with SVM yielding the least precision and accuracy between them [33]. However, custom CNN architectures have been reported to provide over 93% with respect to precision, accuracy, and other metrics within the confusion matrix [30]. Table 3 also advises that research on the RAS inspection and monitoring of aircraft fuselage has demonstrated that CNNs can provide up to 92% accuracy in the detection of surface and joints corrosion [84], while DNNs can provide a better performing accuracy of 96% [85]. In the case of power lines, Table 1 shows us that there is a preference for the use of custom -CNNs, Faster R-CNN or the YOLO v3 technique for extracting, analyzing, and classifying data collected from the RAS inspection of power lines; with these techniques having been reported to provide over 90% in their precision or accuracy in classifying image data [30, 88, 104]. There seems to currently be very little literature on the use machine learning techniques in the support of RAS inspection of vessels, one of the exceptions is research documented by Ortiz et al. [103], where the ANN technique has been used to extract, classify, and analyze corrosion, cracks and coating breakdown from image data collected by a micro-aerial vehicle [103].

Most of the limited literature that examines the application of machine learning techniques supporting the RAS inspection of automobiles focuses on the damage classification of damaged vehicles, because of accidents and associated insurance claims. The research shows a preference for CNNs or Mask R-CNN object recognition or damage detection of automobiles, with CNNs providing accuracy as high as 87% and the Mask R-CNN as high as 94% [95, 96].

Figure 2 provides an illustration of our findings regarding the frequency of use of popular ML techniques to process, analyze and model damage mechanisms on mechanical systems and civil infrastructure.

Table 3 lists and maps the machine learning methods used in the robot inspection of mechanical systems and infrastructure that have been reviewed in this paper. Also, following on from the performance evaluation metrics discussed in Section 4 of this paper, Table 3 provides us with performance evaluation figures and results of the machine learning techniques deployed to process and analyze data collected by RAS platforms for civil mechanical infrastructure in reviewed papers.

6 Technology gaps and challenges

This section reviews technology gaps and challenges in the application of machine learning techniques for robotic inspection of mechanical systems and civil infrastructure.

6.1 Challenges of small object detection for deep learning techniques

Object detection of small (perhaps even undetectable to the human eye) damage mechanisms on mechanical and civil infrastructure has invaluable application to industries ranging from aerospace (for detecting cracks on aircraft) to energy and utility (for detecting erosion or corrosion). A small object has been defined by [106, 107] as a 32 × 32 pixels object within an image. Current literature acknowledges that while object detection of medium to large size objects in image data is now a proven technology, accurate detection of small objects has not yet been mastered and it remains a challenge for researchers [106, 108–111].

The reasons for this research gap are a result of several realities and constraints of current state of the art object detection technology. The first is that small objects are challenging to detect because high-level resolution-based feature maps, that are characteristic of CNN architectures, used to identify large objects in images, do not support the identification of small objects in images. This is because small objects in images are mostly in low resolution. The second is due to currently limited context data; there is significantly less pixels associated with small objects, resulting in little to nothing for the detection algorithms to identify. Furthermore, there is a class imbalance in the datasets that are currently being used to train deep learning models. Current image datasets usually comprise of large to medium sized objects, this results in an imbalance in the groups of object sizes in images available to deep learning models for training.

There is a gap in the research and development of deep learning techniques or models that could provide the higher precision required for accurate localization for

Table 3 Evaluation of Machine Learning techniques used for RAS-based inspection

Machine Learning Techniques	RAS	Applications	Type of Data	Sensors	Performance Evaluation of Technique
DCNN (Deep Convolutional Neural Network)	UAVs (or Drones)	Civil infrastructure such as storage silos, local roadways, etc.	3D Rendering, HD images	LiDAR, High resolution cameras	Gopalakrishnan *et al.* [49] reported that the proposed method can achieve up to 90% accuracy in finding cracks.
Faster R-CNN (Region-Based Convolutional Neural Networks); YOLO; R-FCN; ResNet; ANN	UAVs	Power line inspection	optical images	Cameras	No performance evaluation of ML technique is provided in literature.
Faster RCNN; Advanced image augmentation	Drones	Inspection of leading edge erosion, vortex generator panels with missing teeth, and lightning receptors on wind turbines	Images	Optical cameras	Shihavuddin *et al.* [31] reported that Faster R-CNN is able to achieve 81.10% MAP on detection of four different types of damage such as erosion, vortex generator panel, and lightning receptor on wind turbines. They also reported that a multi-scale pyramid and patching scheme could improve precision by 35%.
Deep convolutional neural networks (DCNN)	Climbing ring robot, Magnetic climbing robot, Multicopter UAV	Wind turbine inspection and maintenance	images	RGB cameras, ZED camera, LiDAR	No performance evaluation of ML technique is provided.
Decision Tree (DT), Support Vector Machine (SVM), and Random Forest (RF)	UAV (Drone)	Pipeline inspection (of cracks, corrosion or erosion)	[i] images [ii] Methane (CH4) [pipeline leakage of gas detector sensor readings] [iii] GPS coordinates	Thermal camera, gas detector, GPS	The paper reports that the Random Forest (RF) algorithm gave the highest accuracy of 97.25% (in terms of measured performance) among other classifiers, with the Support Vector Machine (SVM) algorithm providing the lowest accuracy of 71%. SVM also provided the least precision, recall and F1-score values, with RF providing DT algorithms providing better results.
Custom-CNN	UAV (Drone)	Pipeline inspection (corrosion)	RGB images; video frames	RGB camera	Research evaluated custom-CNN, ZFNet and VGGNET on corrosion pipeline image dataset found performance increased in proportion with the input image size, however the custom-CNN developed by Bastian *et al.* [30] outperformed the ZFNet and VGGNET models, e.g., when provided with an input image 128 × 128, the respective recall(%), precision(%), F1 score (%) and accuracy(%) for the ZFNet model was 83.5, 85.2, 84.34, 85.3; and VGGNet model was 88.0, 92.1, 90.0 and 91.3; with the custom CNN model performing at 93.3, 94.2, 93.75, 94.5 Bastian *et al.* [30].
Custom-CNN	UAV (Drone)	Pipeline inspection (corrosion)	RGB images; video frames	RGB camera	Research evaluated custom-CNN, ZFNet and VGGNET on corrosion pipeline image dataset found performance increased in proportion with the input image size, however the custom-CNN developed by Bastian *et al.* [30] outperformed the ZFNet and VGGNET models e.g. when provided with an input image 128 x 128, the respective recall (%).

Table 3 (*Continued*)

Machine Learning Techniques	RAS	Applications	Type of Data	Sensors	Performance Evaluation of Technique
Modified deep inception neural network (DINN)	Not Applicable	Sub-surface damage inspection of steel bridges	Thermal data (video and images)	Micro-bolometer	Ali *et al.* [100] indicated that the modified DINN developed for their research to analyze cropped thermal images of damages from sub-surface steel bridges achieved a 96% accuracy with the training and testing processes.
DNNs: vgg-f DNN (for feature extraction) and SVM classifier; SURF (key point detector for identification)	Drone	Automatic image detection of defects on aircraft fuselage	jpeg images 3888×5184 resolution of aircraft fuselage	camera	Authors report over 96% accuracy (with onl 3.63% misclassification) at around 15 s processing time for a high-resolution (20-megapixel) image. Malekzadeh *et al.* [85]
CNNs	D-Sight Aircraft Inspection System (DAIS)	Aircraft fuselage surface and joints corrosion monitoring and detection	images	CCD camera	Paper reports precision of over 93%. The CNN architectures respectively used include DenseNet providing the highest accuracy of 92.2% accuracy, SqueezeNet 91.6% accuracy and Iceptionv3 90.6% accuracy.
SSD, R-CNN and YOLO v3	UAVs	Powerline inspection and monitoring for defects	images	camera	Paper indicates YOLO v3 performed the best with 92% precision with relatively few misclassifications.
DCNNs	unspecified vehicle	Automatics detection of rail surface defects	video	video camera	Paper reports the use of small, medium and large DCNNs for rail detection of rail defects, and that rail defects were successfully classified with almost 92% accuracy. The paper reports that the large DCNN model performed better for classification than the small and medium DCNN model.
ANN	micro-aerial vehicle (MAV)	Vessel inspection to detect corrosion, cracks and coating breakdown	images	cameras	No performance evaluation of ML technique is provided.
CNNs	Not applicable	object recognition on damaged (car) vehicle datasets	images	camera	Paper reports using CNN for object recognition on damaged vehicle datasets and obtaining accuracy as high 87.9% during experiments, with customized CNN model yielding 63% accuracy and VGG-16 model yielding 87.9% accuracy.
Mask RCNN	Not applicable	Detection and segmentation of damaged areas on automobiles	images	Camera	Paper reports that the Mask RCNN algorithm provided a 94.53% damage detection accuracy, with an improved mask RCNN algorithm providing for a 96.68% detection accuracy.

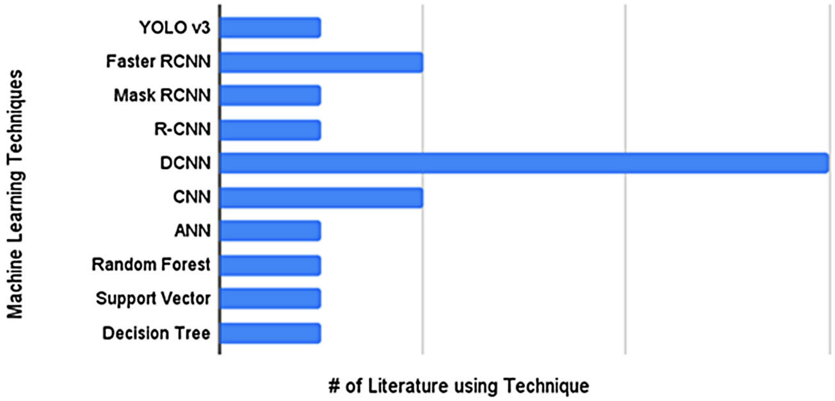

Figure 2 The use of different Machine Learning techniques in literature

small objects in images; as well as an on-going race between researchers in the improvement of current object detection deep learning algorithms for small objects in image datasets [106, 108–111].

6.2 Evaluating accuracy and performance of machine learning techniques

Following this paper's review of the methods and metrics used in academic literature to evaluate the performance of machine learning methods, techniques and models when trained on datasets to output values, interpretations, or predictions; this section briefly reviews critics of these metrics.

There are arguments in literature that contend that the current methods (such as the confusion matrix, accuracy, precision, MAP, RSME and quantile error) used for evaluating the performance and utility of machine learning techniques when applied as solutions to extract or analyze data, et al., can only be understood and applied by subject matter experts in statistics, computer science, artificial intelligence (AI), etc. [78, 112].

Both Shen *et al.* [112] and Beauxis-Aussalet *et al.* [78] contended that non-subject matter experts do not always have the background knowledge to understand terminology such as true negative (TN) or false positive (FP), that form part of the underlying metric framework for evaluating machine learning techniques. Furthermore, Shen *et al*.'s [112] research found that non-experts found it challenging to both use and relate some of the evaluation metrics back to the problem that the techniques are being applied to solve. Beauxis-Aussalet *et al.* [78] underscored the fact that some of the evaluation metrics that exist can even be misunderstood, misinterpreted, or even deployed incorrectly to case studies by non-subject matter experts [78]. It is therefore a contention in literature that there is a gap or requirement for more accessible methods for evaluating the performance of machine learning techniques, that can be understood and used by both subject matter experts in AI and their lay colleagues.

6.3 Machine learning challenges with unstructured data

Throughout the course of this paper, we have reviewed the data collected by RAS systems during inspection of mechanical infrastructure, the types of data collected, the techniques that have been deployed to process and analyze the data collected. This section extends our review of data collected by robotic and autonomous systems to examine the structure of this data and the gaps in our ability to work with this data.

Structured and unstructured is a type of description that data scientists and researchers use to categorize data. Structured data categorizes data with a schema, which means that data is seen as having some sort of logical organization. Structured data is quantitative and is usually displayed as numbers, dates, values, and strings. Structured data can be queried, searched, and analyzed because it is organized in rows and columns, e.g., CSV files, spreadsheets, SQL databases, etc. Traditional sources for structured data vary from sensors, weblogs, network traffic, etc.

Unstructured data however, cannot be contained in rows and columns and has no discernable structure or logic. It is qualitative data, comprising of video, audio, images etc. Structured data cannot be processed or analyzed with the same methods used for structured data, e.g., rows and columns, databases etc.

The challenge for computer scientists and data and AI scientists is that most of the machine learning tools available today, are better suited to train on datasets with structured data. However, most data in the world are unstructured, data in unstructured formats. As indicated by Rai *et al.* [113], literature and research estimate that over 80% of the collected data in the world is unstructured [113, 114]. Traditional sources for unstructured data include social media platforms, images videos and audio data [113].

While there are some AI techniques and tools that can process, analyze, and train unstructured data, e.g., Natural Language Processing (NLP) techniques are used to add structure, such as context and syntax to unstructured text data, while AI techniques such as autoencoders are used to extract and analyze unstructured data. However, these tools are few and the technology has not yet sufficiently matured. This therefore means that there is a gap or requirement for research and development into effective machine learning tools and techniques that can process and analyze unstructured data [113, 114].

6.4 Big data and challenges with real-time data analytics

This paper has reviewed how literature has characterized the data collected by RAS systems in terms of volume, veracity, variety, and velocity. It is also noteworthy to mention the fact that current literature indicates that the vast amounts of volume, variety, and complexity of data being collected by modern day sensors, robotic and autonomous systems has resulted in today's large datasets being referred to as 'Big Data'. The term Big Data refers to a variety of high volume and high velocity datasets, comprising of structured, semi-structured and unstructured data, collected from, and feeding into social networks, academia, sensors networks, international trading markets, surveillance, and communication networks [82, 115–117].

Big Data is exceeding the capacity and capability of current technology to contain, process and analyze in real-time, without the need for data storage and batch processing, in support of real-time dependent applications for international trading markets, smart city infrastructure and robot autonomous systems, e.g. self-driving cars etc. [82, 115, 116, 118]. The very fundamentals of neural networks (deep learning) techniques means that they are best suited for the processing and analysis of Big Data, as neural network algorithms require vast amounts of data to train on to provide meaningful predictions, pattern recognition or representations of any real use. However, while these techniques have resulted in technologies e.g., speech recognition, computer vision and natural language processing (NLP) that can be applied to volumes of unsupervised and unstructured data, however, these technologies are still in their infancy and have not yet matured to the point of coping with complex variety, high volume, and velocity of Big Data [82, 115–118].

6.5 On-board integration of machine learning with RAS platforms

In this paper, we have described robotic platforms as autonomous systems. We have also discussed the machine learning techniques, algorithms that process and analyze the data they collect. However, Panella [119] made an argument that while UAVs are capable of semi-autonomous operation, there is still no integrated unitary technique providing complete autonomy for UAV platforms, that allows for real-time decision-making within the environment they are located within; compared to responding to stimuli or events based on pre-programming. This was one of the stated reasons for the development of an on-board integration of various AI or machine learning techniques to deliver and effect the fully autonomous UAV system that can "think" like humans and make decisions within their environment.

Despite current strides in research and development, there are still challenges in the integration of machine learning techniques on-board RAS platforms. Ono *et al.* [120] noted that while there is a suite of on-board algorithms that can be integrated as part or alongside the Robot Operating System (ROS) providing robotic systems with the autonomy to respond to events in their environment and complete tasks (e.g., the Mars rover); there is still a gap in available algorithms and technology that could provide complete on-board autonomy for future rover missions [120].

Furthermore, Ono *et al.* [120] discussed about the gaps in intelligent algorithms on-board robot systems which could result in what the paper refers to as the "unnoticed green monster problem" where human decision-makers and operators are not able to take real-time action to events or stimuli detected by the RAS system (the mars rover in this particular case study), due to delay or loss of data (imagery or otherwise), being fed from the robot system, e.g., Mars to the human operator, in this case, a control centre on Earth. The point being that this demonstrated the need for the development and on-board integration of AI algorithms that would provide on-board decision-making on the robot platform, to enable real-time responses for what Ono *et al.* [120] describes as "scientific opportunities and avoid the "green monster problem" [120].

Hillebrand *et al.* [121] and Contreras *et al.* [122] suggested the use of deep learning (neural networks), specifically reinforcement learning, as a response to the absence of neural network design methodology in robotic systems [121, 122]. Chen *et al.* [123] noted that while autonomous robot navigation exists as a mainstream technology, the current capability still has challenges in its ability to manage complex and dynamic environments and reduce misclassifications by current perception algorithms [123].

7 Conclusions

This review has reported on the types of robotic platforms deployed for inspection of different mechanical systems and civil infrastructure such as storage tanks, high rise facilities and nuclear power plants. While unmanned marine vehicles are deployed for systems located underwater (such as subsea power cables); unmanned ground robots are better suited to horizontal ground surface environments, with UAVs mostly deployed for both indoor and outdoor, remote, hazardous environments.

This paper demonstrated through an extensive literature review that machine learning, has been used with varied efficacy, to support the processing and analysis and classification of data collected by RAS systems during inspection of mechanical and civil infrastructure. The review revealed that there are few studies demonstrating use of deep learning techniques for the analysis of datasets collected during structural health inspections. In these studies, it was shown that deep learning techniques performed better than most machine learning methods in the processing and analysis of image (damage mechanism) datasets. Furthermore, almost all research reviewed have focused on the inspection, analysis, and classification of single damage mechanisms, e.g., corrosion, cracks, erosion, etc. This indicated a research gap in the use and application of machine learning techniques to analyze and classify multiple types of damage mechanisms from video or image datasets collected during the inspection of mechanical systems and civil infrastructure.

Funding
This research was funded by EPSRC https://gtr.ukri.org/projects?ref=studentship-2466663#/tabOverview, grant number 2466663.

Availability of data and materials
The authors confirm that the data supporting the findings of this study are available within the article and its supplementary materials.

Code availability (software application or custom code)
Not applicable.

Declarations

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
"Conceptualization, MM and MS; methodology, MM and MS; investigation, MM; resources, MM; MS; writing—original draft preparation, MM and MS; writing—review and editing, MM and MS; supervision, MS; project administration, MM; funding acquisition, MS. All authors read and approved the final manuscript.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 7 January 2022 Accepted: 5 April 2022
Published online: 29 April 2022

References

1. H.M. La, N. Gucunski, K. Dana, S.-H. Kee, Development of an autonomous bridge deck inspection robotic system. J. Field Robot. **2017**, 1489 (2017)
2. L. Wang, Z. Zhang, Automatic detection of wind turbine blade surface cracks based on UAV-taken images. IEEE Trans. Ind. Electron. **64**(9), 7293–7303 (2017)
3. S. Bernardini, F. Jovan, Z. Jiang, S. Watson, A. Weightman, P. Moradi, T. Richardson, R. Sadeghian, S. Sareh, A multi-robot platform for the autonomous operation and maintenance of offshore wind farms blue sky ideas track, in *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS 2020*, May 9–13, 2020, Auckland, New Zealand (2020)
4. C. Stout, D. Thompson, UAV Approaches to Wind Turbine Inspection: Reducing Reliance on Rope-Access. Offshore Renewable Energy Catapult. (2019)
5. D. Schmidt et al., Climbing robots for maintenance and inspections of vertical structures—A survey of design aspects and technologies. Robot. Auton. Syst. (2013). https://doi.org/10.1016/j.robot.2013.09.002
6. D. Lattanzi et al., Review of Robotic Infrastructure Inspection Systems. J. Infrastruct. Syst. (2017). https://doi.org/10.1061/(ASCE)IS.1943-555X.0000353
7. M.A.M. Yusoff et al., Development of a Remotely Operated Vehicle (ROV) for underwater inspection. Jurutera (2013)
8. A.L. Meyrowitz et al., Autonomous vehicles, in *Proceedings of the IEEE 1996* (1996). https://doi.org/10.1109/5.533960
9. F. Rubio et al., A review of mobile robots: Concepts, methods, theoretical framework, and applications. Int. J. Adv. Robot. Syst. **2019** (2019). https://doi.org/10.1177/1729881419839596
10. D.W. Gage, A Brief History of Unmanned Ground Vehicle (UGV) Development Efforts (1995)
11. W. Shen et al., Proposed wall climbing robot with permanent magnetic tracks for inspecting oil tanks, in *IEEE International Conference Mechatronics and Automation* (2005). https://doi.org/10.1109/ICMA.2005.1626882
12. L.P. Kalra et al., A wall climbing robot for oil tank inspection, in *2006 IEEE International Conference on Robotics and Biomimetics* (2006). https://doi.org/10.1109/ROBIO.2006.340155
13. S. Campbell et al., Sensor technology in autonomous vehicles: a review, in *2018 29th Irish Signals and Systems Conference*, ISSC, 2018 (2018). https://doi.org/10.1109/ISSC.2018.8585340
14. J. Seo et al., Drone-enabled bridge inspection methodology and application. Autom. Constr. (2018). https://doi.org/10.1016/j.autcon.2018.06.006. https://www.sciencedirect.com/science/article/pii/S0926580517309755 DOI
15. M. Shafiee et al., Unmanned Aerial Drones for Inspection of Offshore Wind Turbines: A Mission-Critical Failure Analysis. Robotics J. (2021). https://doi.org/10.3390/robotics10010026
16. M.H. Frederiksen et al., Drones for inspection of infrastructure: Barriers, opportunities and successful uses. Center for Integrative Innovation Management (2019)
17. M. Drones Lt, Best Commercial Drones for Beginners, Sep. 02, 2019, 2018. https://www.coptrz.com/best-commercial-drones-for-beginners/
18. C. Eschmann et al., High-resolution multisensor infrastructure inspection with unmanned aircraft systems, in *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences 2013* (2013). https://doi.org/10.5194/isprsarchives-XL-1-W2-125-2013 https://ui.adsabs.harvard.edu/abs/2013ISPAr.XL1b.125E
19. X.L. Ding et al., A review of structures, verification, and calibration technologies of space robotic systems for on-orbit servicing (2020). https://doi.org/10.1007/s11431-020-1737-4
20. A. Flores-Abad et al., *A Review of Space Robotics Technologies for on-Orbit Servicing* (Elsevier, Amsterdam, 2014). https://doi.org/10.1016/j.paerosci.2014.03.002
21. P.J. Staritz et al., Skyworker: A Robot for Assembly, Inspection and Maintenance of Large-Scale Orbital Facilities. IEEE (2001). https://doi.org/10.1109/ROBOT.2001.933271
22. H. Choset, D. Kortenkamp, Path planning and control for free-flying inspection robot in space. J. Aerosp. Eng. (1999). https://doi.org/10.1061/(ASCE)0893-1321(1999)12:2(74)
23. J.S. Mehling et al., A minimally invasive tendril robot for in-space inspection, in *The First IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, BioRob 2006* (2006), pp. 690–695. https://doi.org/10.1109/BIOROB.2006.1639170
24. S.-I. Nishida et al., Prototype of an end-effector for a space inspection robot. Adv. Robot. (2012). https://doi.org/10.1163/156855301300235788
25. L. Pedersen et al., A survey of space robotics, in *ISAIRAS* (2003)

26. J. Redmon et al., You only look once: unified, real-time object detection, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016)
27. W. Fan et al., Mining big data: current status, and forecast to the future, in *2013 Association for Computing Machinery* (2013). https://doi.org/10.1145/2481244.2481246
28. B. Matturdi et al., Big data security and privacy: a review. China Commun. **11**(14), 135–145 (2014). https://doi.org/10.1109/CC.2014.7085614
29. D. Laney, *3-D Data Management: Controlling Data Volume, Velocity and Variety*. META Group Research Note, February, vol. 6 (2001)
30. B.T. Bastian, J. N, S.K. Kumar, C.V. Jiji, Visual inspection and characterization of external corrosion in pipelines using deep neural network. NDT & E International Journal **107**, 102134 (2019)
31. A. Shihavuddin et al., Wind turbine surface damage detection by deep learning aided drone inspection analysis. Energies **12**(4), 676 (2019). https://doi.org/10.3390/en12040676
32. M. Hassanalian et al., Classifications, applications, and design challenges of drones: a review. Prog. Aerosp. Sci. (2017). https://doi.org/10.1016/j.paerosci.2017.04.003. https://www.sciencedirect.com/science/article/pii/S0376042116301348
33. A. Alharam et al., Real time AI-based pipeline inspection using drone for oil and gas industries in Bahrain, in *2020 International Conference on Innovation and Intelligence for Informatics, Computing and Technologies (3ICT)* (2020)
34. V. Nasteski, An overview of the supervised machine learning methods. Horizons B **4** (2017). https://doi.org/10.20544/HORIZONS.B.04.1.17.P05
35. B. Mahesh, *Machine Learning Algorithms – a Review* (2019). https://doi.org/10.21275/ART20203995
36. A. Carrio et al., A review of deep learning methods and applications for unmanned aerial vehicles. Hindawi J. Sens. (2017). https://doi.org/10.1155/2017/3296874
37. M.N. Mohammed et al., Design and Development of Pipeline Inspection Robot for Crack and Corrosion Detection (2018)
38. https://www.analyticssteps.com/blogs/how-does-k-nearest-neighbor-works-machine-learning-classification-problem
39. F. Hoffmann et al., Benchmarking in classification and regression. WIREs Data Mining Knowl. Discov. **9**, e1318 (2019). https://doi.org/10.1002/widm.1318
40. A. Geron, Hands-On Machine Learning with Scikit-Learn, Keras &TensorFlow, 2nd edn. (2019). 2019
41. I. Goodfellow et al., *Deep Learning* (MIT Press, Cambridge, 2016)
42. F.Y. Osisanwo et al., Supervised machine learning algorithms: classification and comparison. Int. J. Comput. Trends. Technol. (IJCTT) **48**(3) 128–138 (2017)
43. C.-F. Tsai et al., Intrusion detection by machine learning: a review. Expert Syst. Appl. **36**(10), 11994–12000 (2009)
44. A. Matsunaga et al., On the use of machine learning to predict the time and resources consumed by applications, in *2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing* (2010), pp. 495–504. https://doi.org/10.1109/CCGRID.2010.98
45. M. Jogin et al., Feature extraction using Convolution Neural Networks (CNN) and deep learning, in *2018 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)* (2018), pp. 2319–2323. https://doi.org/10.1109/RTEICT42901.2018.9012507
46. https://ujjwalkarn.me/2016/08/09/quick-intro-neural-networks/
47. https://morioh.com/p/73fce91e9846
48. Y. Guo et al., Deep learning for visual understanding: a review. Neurocomputing **187**, 27–48 (2016). https://doi.org/10.1016/j.neucom.2015.09.116
49. K. Gopalakrishnan et al., Crack damage detection in unmanned aerial vehicle images of civil infrastructure using pre-trained deep learning model. Int. J. Traffic Transp. Eng. (IJTTE) (2017)
50. H. Larochelle et al., Exploring strategies for training deep neural networks. J. Mach. Learn. Res. **1**, 1–40 (2009). https://doi.org/10.1145/1577069.1577070
51. A. Fischer, C. Igel, *An Introduction to Restricted Boltzmann Machines*. Iberoamerican Congress on Pattern Recognition (Springer, Berlin, 2012)
52. N. Agarwalla et al., Deep learning using restricted Boltzmann machines. Int. J. Comput. Sci. Inf. Secur. **7**(3), 1552–1556 (2016)
53. Y. Hua et al., Deep belief networks and deep learning, in *Proceedings of 2015 International Conference on Intelligent Computing and Internet of Things* (2015), pp. 1–4. https://doi.org/10.1109/ICAIOT.2015.7111524
54. https://blog.paperspace.com/faster-r-cnn-explained-object-detection/
55. https://neurohive.io/en/popular-networks/r-cnn/
56. Z.-Q. Zhao et al., Object detection with deep learning: a review. IEEE Trans. Neural Netw. Learn. Syst. **30**(11), 3212–3232 (2019). https://doi.org/10.1109/TNNLS.2018.2876865
57. https://blog.athelas.com/a-brief-history-of-cnns-in-image-segmentation-from-r-cnn-to-mask-r-cnn-34ea83205de4
58. P.S. Bithas et al., A Survey on Machine-Learning Techniques for UAV-Based Communications. Sensors (Basel, Switzerland) **26 November 2019** (2019). https://europepmc.org/articles/PMC6929112. Accessed September 2020
59. R. Girshick et al., Rich feature hierarchies for accurate object detection and semantic segmentation, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2014)
60. R. Girshick, Fast r-cnn, in *Proceedings of the IEEE International Conference on Computer Vision* (2015)
61. K. He et al. Mask R-CNN. In ICCV, 2017
62. J. Dai et al., R-FCN: Object Detection via Region-based Fully Convolutional Networks (2016). arXiv:1605.06409
63. W. Liu et al., Ssd: Single shot multibox detector (2015). Preprint arXiv:1512.02325
64. S. Ren et al., Faster r-cnn: Towards real-time object detection with region proposal networks. Adv. Neural Inf. Process. Syst. **2015** (2015)
65. S. Grossberg, Recurrent neural networks. Scholarpedia **8**(2), 1888 (2013)
66. J.A. Bullinaria, Recurrent neural networks. Neural Computation: Lecture 12 (2013)
67. S. Hochreiter, J. Schmidhuber, Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997). https://doi.org/10.1162/neco.1997.9.8.1735
68. I.J. Goodfellow et al., Generative Adversarial Networks (2014). arXiv, stat.ML
69. M. Mirza, S. Osindero, Conditional Generative Adversarial Nets (2014)
70. L. Noriega, Multilayer perceptron tutorial. School of Computing. Staffordshire University (2005)
71. H. Taud, J. Mas, Multilayer perceptron (MLP), in *Geomatic Approaches for Modeling Land Change Scenarios*. Lecture Notes in Geoinformation and Cartography. (Springer, Cham, 2018). https://doi.org/10.1007/978-3-319-60801-3_27
72. G. Alain, Y. Bengio, What Regularized Auto-Encoders Learn from the Data Generating Distribution (2014)
73. F.Q. Lauzon, An introduction to deep learning, in *2012 11th International Conference on Information Science, Signal Processing and Their Applications (ISSPA)*, (2012), pp. 1438–1439. https://doi.org/10.1109/ISSPA.2012.6310529
74. P. Baldi, Autoencoders, unsupervised learning, and deep architectures, in *Proceedings of ICML Workshop on Unsupervised and Transfer Learning* (2012)
75. Q.V. Le, A tutorial on deep learning part 2: autoencoders, convolutional neural networks and recurrent neural networks. Google Brain **20**, 1–20 (2015)
76. A. Agarwal, A. Motwani, *An Overview of Convolutional and AutoEncoder Deep Learning Algorithm* (2016)
77. Y. Coadou, Boosted decision trees and applications. EPJ Web Conf. **55**, 02004 (2013). https://doi.org/10.1051/epjconf/20135502004
78. E. Beauxis-Aussalet et al., Visualization of confusion matrix for non-expert users, in *IEEE Conference on Visual Analytics Science and Technology (VAST) - Poster Proceedings* (2014)
79. G. Shobha et al., Machine learning, in *Handbook of Statistics*, vol. 38 (Elsevier, Amsterdam, 2018), pp. 197–228. https://doi.org/10.1016/bs.host.2018.07.004. https://www.sciencedirect.com/science/article/pii/S0169716118300191. ISSN 0169-7161. ISBN 9780444640420
80. A. Kulkarni et al., Foundations of data imbalance and solutions for a data democracy, in *Data Democracy* (Academic Press, San Diego, 2020), pp. 83–106. https://doi.org/10.1016/B978-0-12-818366-3.00005-8. ISBN 9780128183663
81. https://towardsdatascience.com/map-mean-average-precision-might-confuse-you-5956f1bfa9e2
82. N. Mohamed et al., Real-time big data analytics: applications and challenges, in *Proc. Int. Conf. High Perform. Comput. Simulation* (2014), pp. 305–310
83. J. Franko et al., Design of a multi-robot system for wind turbine maintenance. Energies (2020)
84. B. Brandoli et al., Aircraft fuselage corrosion detection using artificial intelligence. Sensors **2021**(21), 4026 (2021). https://doi.org/10.3390/s21124026

85. T. Malekzadeh et al., Aircraft Fuselage Defect Detection using Deep Neural Networks (2017). arXiv:1712.09213
86. J. Miranda et al., Machine learning approaches for defect classification on aircraft fuselage images aquired by an UAV, in *Fourteenth International Conference on Quality Control by Artificial Vision*. 16 July 2019, Proc. SPIE, vol. 11172 (2019), p. 1117208. https://doi.org/10.1117/12.2520567.
87. B. Jalil et al., Fault detection in power equipment via an unmanned aerial system using multi modal data. Sensors **2019**(19), 3014 (2019). https://doi.org/10.3390/s19133014
88. E. Titov et al., The deep learning based power line defect detection system built on data collected by the cablewalker drone, in *2019 International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON)* (2019), pp. 0700–0704. https://doi.org/10.1109/SIBIRCON48586.2019.8958397
89. A. Ortiz et al., First steps towards a roboticized visual inspection system for vessels, in *2010 IEEE 15th Conference on Emerging Technologies & Factory Automation (ETFA 2010)* (2010), pp. 1–6. https://doi.org/10.1109/ETFA.2010.5641246
90. F. Bonnin-Pascual et al., Semi-autonomous visual inspection of vessels assisted by an unmanned micro aerial vehicle, in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems* (2012), pp. 3955–3961. https://doi.org/10.1109/IROS.2012.6385891
91. S. Kawabata et al., Autonomous flight drone with depth camera for inspection task of infra structure, in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 2 (2018)
92. Y.-J. Cha et al., Deep learning-based crack damage detection using convolutional neural networks. Comput.-Aided Civ. Infrastruct. Eng. **32**, 361–378 (2017)
93. M.M. Karim et al., Modeling and simulation of a robotic bridge inspection system, in *Procedia Computer Science* (2020), pp. 177–185. https://doi.org/10.1016/j.procs.2020.02.276. https://www.sciencedirect.com/science/article/pii/S1877050920304154. ISSN 1877-0509
94. M.A. Manzoor et al., Vehicle make and model classification system using bag of SIFT features, in *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*, (2017), pp. 1–5. https://doi.org/10.1109/CCWC.2017.7868475
95. P. Rakshata et al., Car damage detection and analysis using deep learning algorithm for automotive. Int. J. Sci. Technol. Res. **5**(6) (2019). Nov-Dec-2019, ISSN (Online): 2395-566X
96. Q. Zhang et al., Vehicle-damage-detection segmentation algorithm based on improved mask RCNN. IEEE Access **8**, 6997–7004 (2020). https://doi.org/10.1109/ACCESS.2020.2964055
97. H. Bandi et al., Assessing car damage with convolutional neural networks, in *2021 International Conference on Communication Information and Computing Technology (ICCICT)* (2021), pp. 1–5. https://doi.org/10.1109/ICCICT50803.2021.9510069
98. C. Giovany Pachón-Suescún et al., Scratch Detection in Cars Using a Convolutional Neural Network by Means of Transfer Learning. IJAER (2018) 16 Nov 2018
99. K. Patil et al., Deep learning based car damage classification, in *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)* (2017), pp. 50–54. https://doi.org/10.1109/ICMLA.2017.0-179
100. R. Ali et al., Subsurface damage detection of a steel bridge using deep learning and uncooled micro-bolometer. Constr. Build. Mater. **226** (2019). https://doi.org/10.1016/j.conbuildmat.2019.07.293. 2019, 376-387, ISSN 0950-0618. https://www.sciencedirect.com/science/article/pii/S0950061819319671
101. Y. Liu et al., The method of insulator recognition based on deep learning, in *Proceedings of the 2016 4th International Conference on Applied Robotics for the Power Industry (CARPI)*, Jinan, China, 11–13 October, 2016 (2016), pp. 1–5
102. Z. Zhao et al., Multi-patch deep features for power line insulator status classification from aerial images, in *2016 International Joint Conference on Neural Networks (IJCNN), 2016* (2016), pp. 3187–3194. https://doi.org/10.1109/IJCNN.2016.7727606
103. A. Ortiz et al., Vision-based corrosion detection assisted by a micro-aerial vehicle in a vessel inspection application. Sensors **2016**(16), 2118 (2016). https://doi.org/10.3390/s16122118
104. V.N. Nguyen et al., Automatic autonomous vision-based power line inspection: A review of current status and the potential role of deep learning. Int. J. Electr. Power Energy Syst. **2018** (2018)
105. S. Faghih-Roohi et al., Deep convolutional neural networks for detection of rail surface defects, in *Neural Networks (IJCNN), 2016 International Joint Conference on, 2016* (2016), pp. 2584–2589
106. Z.-Q. Tong et al., Recent advances in small object detection based on deep learning: a review. Image Vis. Comput. **97**, 103910 (2020)
107. T.-Y. Lin et al., *Microsoft Coco: Common Objects in Context*. European Conference on Computer Vision (Springer, Cham, 2014)
108. Y. Liu et al., A survey and performance evaluation of deep learning methods for small object detection. Expert Syst. Appl. **172**, 114602 (2021)
109. N.-D. Nguyen et al., An evaluation of deep learning methods for small object detection. J. Electr. Comput. Eng. **2020**, 3189691 (2020)
110. C. Chenyi et al., R-CNN for small object detection, in *Asian Conference on Computer Vision* (Springer, Cham, 2016)
111. Z.-Q. Zhao et al., Object detection with deep learning: a review. IEEE Trans. Neural Netw. Learn. Syst. **30**(11), 3212–3232 (2019)
112. Y. Shen et al., Design alternative representations of confusion matrices to support non-expert public understanding of algorithm performance. Proc. ACM Hum. Comput. Interact. **4**(CSCW2), 153 (2020)
113. R.K. Rai et al., Intricacies of Unstructured Data. EAI Endorsed Transactions on Scalable Information Systems **4**(14) (2017). https://doi.org/10.4108/eai.25-9-2017.153151
114. A. Gandomi et al., Beyond the hype: big data concepts, methods, and analytics. Int. J. Inf. Manag. **35**(2), 137–144 (2015)
115. D.P. Acharjya et al., A survey on big data analytics: challenges, open research issues and tools. Int. J. Adv. Comput. Sci. Appl. **7**, 511–518 (2016)
116. A. Oussous et al., Big data technologies: a survey. J. King Saud Univ, Comput. Inf. Sci. **30**, 431–448 (2018)
117. X. Jin et al., Significance andchallenges of big data research. Big Data Res. **2**(2), 59–64 (2015)
118. M.M. Najafabadi et al., Deep learning applications and challenges in big data analytics. Big Data **2**(1), 1–21 (2015)
119. I. Panella, Artificial intelligence methodologies applicable to support the decision-making capability on board unmanned aerial vehicles, in *ECSIS Symposium on Bio-Inspired Learning and Intelligent Systems for Security*, Edinburgh (2008), pp. 111–118. https://doi.org/10.1109/BLISS.2008.14
120. M. Ono et al., MAARS: machine learning-based analytics for automated rover systems, in *Proc. IEEE Aerosp. Conf* (2020), pp. 1–17
121. M. Hillebrand et al., A design methodology for deep reinforcement learning in autonomous systems. Procedia Manufacturing **52** (2020). https://doi.org/10.1016/j.promfg.2020.11.044. https://www.sciencedirect.com/science/article/pii/S2351978920321879
122. S. Contreras et al., Using deep learning for exploration and recognition of objects based on images, in *2016 XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR)* (2016), pp. 1–6. https://doi.org/10.1109/LARS-SBR.2016.8
123. W. Chen et al., Door recognition and deep learning algorithm for visual based robot navigation, in *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO 2014)* (2014), pp. 1793–1798. https://doi.org/10.1109/ROBIO.2014.7090595