Article

Using Convolutional Neural Networks to Automate Aircraft Maintenance Visual Inspection

Anil Doğru [1], Soufiane Bouarfa [2,3], Ridwan Arizar [4], and Reyhan Aydoğan [1,5]*

[1] Computer Science, Özyegin University; anil.dogru@ozu.edu.tr
[2] Delft Aviation; soufiane@delftaviation.com
[3] Abu Dhabi Polytechnic; soufiane.bouarfa@adpoly.ac.ae
[4] Singular Solutions B.V.; r.arizar@singulairsolutions.com
[5] Interactive Intelligence Group, Delft University of Technology; reyhan.aydogan@ozyegin.edu.tr
* Correspondence: soufiane@delftaviation.com

Abstract: Convolutional Neural Networks combined with autonomous drones are increasingly seen as enablers of partially automating the aircraft maintenance visual inspection process. Such an innovative concept can have a significant impact on aircraft operations. Through supporting aircraft maintenance engineers detect and classify a wide range of defects, the time spent on inspection can significantly be reduced. Examples of defects that can be automatically detected include aircraft dents, paint defects, cracks and holes, and lightning strike damage. Additionally, this concept could also increase the accuracy of damage detection and reduce the number of aircraft inspection incidents related to human factors like fatigue and time pressure. In our previous work, we have applied a recent Convolutional Neural Network architecture known by MASK R-CNN to detect aircraft dents. MASK-RCNN was chosen because it enables the detection of multiple objects in an image while simultaneously generating a segmentation mask for each instance. The previously obtained F_1 and F_2 scores were 62.67% and 59.35 % respectively. This paper extends the previous work by applying different techniques to improve and evaluate prediction performance experimentally. The approaches uses include (1) Balancing the original dataset by adding images without dents; (2) Increasing data homogeneity by focusing on wing images only; (3) Exploring the potential of three augmentation techniques in improving model performance namely flipping, rotating, and blurring; and (4) using a pre-classifier in combination with MASK R-CNN. The results show that a hybrid approache combining MASK R-CNN and augmentation techniques leads to an improved performance with an F_1 score of (67.50%) and F_2 score of (66.37%).

Keywords: Aircraft Maintenance Inspection; Anomaly Detection; Defect Inspection; Convolutional Neural Networks; Mask R-CNN; Generative Adversarial Networks; Image Augmentation

1. Introduction

1.1. Automated Aircraft Maintenance Inspection

Automated aircraft inspection basically aims at automating the visual inspection process normally carried out by aircraft engineers. It aims at detecting defects that are visible on the aircraft skin which are usually structural defects [1]. These defects can include dents, lightning strike damage, paint defects, fasteners defects, corrosion, cracks, just to name a few. Automatic defect detection can be enabled by using a drone-based system that can scan the aircraft and detect/ classify a wide range of defects in a very short time. Other alternatives would be using sensors in a smart hangar or at the airport apron area. Automating the visual aircraft inspection process can have a significant impact on today's flight operations with numerous benefits including but not limited to:

- **Reduction of inspection time and AOG time:** The sensors either on-board a drone or in a smart hangar can quickly reach difficult places such as the flight control surfaces in both wings and the

 doi:10.20944/preprints202011.0527.v1

empennage. This in turn can reduce the man hours and preparation time as engineers would need heavy equipment such as cherry pickers to have more scrutiny. The inspection time can be even further reduced if the automated inspection system is able to assess the severity of the damage and the affected aircraft structure with reference to both aircraft manuals (AMM and SRM), and recommend the course of action to the engineers. Time savings on inspection time would consequently lead to reductions of up to 90% in Aircraft-On-Ground times [2].

- **Reduction of safety incidents and PPE related costs:** Engineers would no longer need to work at heights or expose themselves to hazardous areas e.g. in case of dangerous aircraft conditions or the presence of toxic chemicals. This would also lead to important cost savings on Personal Protective Equipment.
- **Reduction of decision time:** Defect detection will be much more accurate and faster compared to the current visual inspection process. For instance, it takes operators between 8 and 12 hours to locate lightning strike damage using heavy equipment such as gangways and cherry-pickers. This can be reduced by 75% if an automated drone-based system is used [3]. Such time savings can free up aircraft engineers from dull tasks and make them focus on more important tasks. This is especially desired given the projected need of aircraft engineers in various regions of the world which is 769000 for the period 2019-2038 according to a recent Boeing study [4].
- **Objective damage assessment and reduction of human error:** If the dataset used by the neural network is annotated by a team of experts who had to reach consensus on what is a damage and what not, then detection of defects will be much more objective. Consequently, the variability of performance assessments by different inspectors will be significantly reduced. Furthermore, human errors such as failing to detect critical damage (for instance due to fatigue or time pressure) will be prevented. This is particularly important given the recurring nature of such incidents. For instance, the Australian Transport Safety Bureau (ATSB) recently reported a serious incident in which a significant damage to the horizontal stabilizer went undetected during an inspection, and was only identified 13 flights later [5]. In [1], it was also shown that the model is able to detect dents which were missed the by experts during the annotations process.
- **Augmentation of Novices Skills:** It takes a novice 10000 hours to become an experienced inspector. Using a decision-support system that has been trained to classify defects on a large database can significantly augment the skills of novices.

1.2. Applications/Breakthroughs of Computer Vision

Computer vision is changing the field of visual assessment in nearly every domain. This is not surprising given the rapid advances and growing popularity of the field. For instance, the error in object detection by a machine decreased from 26% in 2011 to only 3% in 2016 which is less than human error reported to be 5% [6].The main driver behind these improvements is deep learning which had a profound impact on robotic perception following the design of AlexNet in 2012. Image classification has therefore become a relatively easy problem to solve given that enough data is available to training the deep learning model.

Computer vision has been successfully applied in the **Healthcare** domain. It typically deals with tasks like object classification, detection, and segmentation which are crucial in determining for instance whether a patient's radiograph has a malignant tumor [7]. Many studies have demonstrated promising results in complex diagnostics in a wide range of areas including dermatology [8], radiology [9], ophthalmology [10], and pathology [11]. In fact, the technology has become so good in medical imaging diagnosis that the FDA has recently approved many use cases [12] such as:

- Arterys MICA from Arterys Inc. which detects liver and lung cancer on CT and MRI (Approved on September 2018).
- HealthPNX from Zebra Medical Vision Ltd. which alerts for pneumothorax based on Chest X-Rays (Approved on May 2019).
- BriefCase from Aidoc Medical, Ltd which identifies linear lucencies in the cervical spine bone in patterns compatible with fractures (Approved on July 2018).

 doi:10.20944/preprints202011.0527.v1

- Critical Care Suite from GE Medical Systems, LLC which identifies pneumothorax based on Chest X-Rays (Approved on August 2019).
- QuantX from Quantitative Insights, Inc. which detects breast cancer (Approved on January 2020).

All these deep learning systems could support physicians by offering them a second opinion and flagging concerning areas and abnormalities in images.

Agriculture is another popular domain where computer vision solutions combined with deep learning algorithms are integrated into drones that can scan large fields in a matter of minutes. Images are collected and processed to help farmers make informed decisions about their crops. The captured images include soil and crop conditions to monitor for any stress or disease. Patricio & Rieder [13] provide a systemic review of 25 papers that treat aspects related to disease detection, grain quality and phenotyping of the most produced grains in the world. The classifiers used by the different research groups include Support Vector Machines, Artificial Neural Networks, Deep Belief Networks, Back-Propagation Neural Networks. Tian *et al.* [14] recently analyzed the body of work of computer vision applications in agriculture and classified it into six main categories:

- Crop health growth monitoring.
- Prevention and control of crop diseases, pests and weeds.
- Automatic harvesting of crops.
- Agricultural product quality testing.
- Modern farm automation management.
- Monitoring of farmland information with UAV.

The authors conclude that these efforts contribute to the development of agricultural automation with all the expected benefits: low cost, high efficiency, and high precision. However, several challenges have also been identified which include 1) the lack of a large scale dataset; 2) the increasing need to integrate more disciplines and agricultural requirements; and 3) ensuring the robustness and accuracy in various complex situations. In another closely related work, Ganesh *et al.* 2019 [15] use mask R-CNN to detect individual fruits and obtain pixel-wise mask for each detected fruit in the image.

Computer vision solutions have also been widely explored in **production and manufacturing** environments to inspect product quality and detect defects. For instance, Yun *et al.* 2020 [16] propose an automatic vision-based defect inspection system to inspect metal surface defects. Because such defects occur rarely, the researchers had an imbalanced data problem. Therefore, they proposed to use convolutional variational autoencoders to generate sufficient data to train the model. Ren *et al.* 2018 [17] also recognize that automating surface inspection is a challenging task as collecting data is usually costly, and have proposed a generic approach that require small training data for automated inspection. The approach builds a classifier on the features of image patches, where the features are transferred from a pre-trained deep learning network. Then, a pixel-wise prediction is obtained by convolving the trained classifier over input image. In another work, Weimer *et al.* 2016 [18] examine different design configurations of deep convolutional neural networks, and the impact of different hyper-parameter settings toward defect detection accuracy. Other applications of computer vision can be found in domains such as the **automotive** industry [19], **retail** [20], and **railway** [21].

The Applications of computer vision and deep learning in **aircraft maintenance inspection** remain very limited despite the impact this field is already making in other domains. Based on the literature and technology review performed by the authors, it was found that only a few researchers and organizations are working on automating aircraft visual inspection.

One of the earliest works that uses neural networks to detect aircraft defects dates back to 2017. In this work [22] the authors used dataset images of the airplane fuselage. For each image, a binary mask was created by an experienced aircraft engineer to represent defects. The authors have used a convolutional neural network that was pre-trained on ImageNet as a feature extractor. The proposed algorithm achieves about 96.37 % accuracy. A key challenge faced by the authors was an imbalanced

 doi:10.20944/preprints202011.0527.v1

dataset which had very few defect photos. To tackle this problem, the authors used data balancing techniques to oversample the rare defect data and undersample the no-defect data.

Miranda *et al.* [23] use object detection to inspect airplane exterior screws with a UAV. Convolutional Neural Networks are used to characterize zones of interest and extract screws from the images. Then computer vision algorithms are used to assess the status of each screw and detect missing and loose ones. In this work, the authors made use of GANs to generate screw patterns using a bipartite approach.

Miranda *et al.* [24] point out to the challenge of detecting rare classes of defects given the extreme imbalance of defect datasets. For instance, there is an unequal distribution between different classes of defects. So, the rarest and most valuable defect samples represent few elements among thousands of annotated objects. To address this problem, the authors propose a hybrid approach which combines classic deep learning models and few-shot learning approaches such as matching network and prototypical network which can learn from few samples. In [25], the authors extend this work by questioning the interface between models in such a hybrid architecture. It was shown that by carefully selecting the data from the well-represented class when using few-shot learning techniques, it is possible to enhance the previously proposed solution.

1.3. Research Objective

In Bouarfa *et al.* [1], we have applied MASK R-CNN to detect aircraft dents. MASK-RCNN was chosen because it enables the detection of multiple objects in an image while simultaneously generating a segmentation mask for each instance. The previously obtained F_1 and F_2 scores were 62.67% and 59.35 % respectively. This paper extends the previous work by applying different techniques to improve and evaluate prediction performance experimentally. The approaches uses include (1) Balancing the original dataset by adding images without dents; (2) Increasing data homogeneity by focusing on wing images only; (3) Exploring the potential of three augmentation techniques in improving model performance namely flipping, rotating, and blurring; and (4) using a pre-classifier in combination with MASK R-CNN.

This paper is organized as follows. Section 1 provides the introduction. Section 2 describes the methodology. Section 3 describes the experimental set-up and presents the key results. The conclusion is provided in section 4.

2. Methodology

This study uses Mask Region Convolutional Neural Networks (Mask R-CNN) to automatically detect aircraft dents. **Mask R-CNN** is a deep learning algorithm for computer vision that can identify multiple objects in one image. The approach goes beyond a plain vanilla CNN in that it allows the the exact location and identification of objects of interest in their bounding.This functionality is relevant for detecting aircraft dents which don't have a clear defined shape. However Mask R-CNN comes at a computational cost. For example, YOLO [26] a popular object detection algorithm is much faster if all what needed are bounding boxes. Another drawback of Mask R-CNN is labelling the masks. Annotating data for mask is a cumbersome and tedious process as the data labeler needs to draw polygon for each of the object in an image.

2.1. Object detection

As with every object detection task, there exist three sub-tasks [27] (see also Figure 1):

- *Extracting Regions of Interest:* The image is passed to a ConvNet which returns the Region of Interests (RoIs) based on methods like selective search (R-CNN) or RPN (Region Proposal Network for faster R-CNN). Then, a pooling layer is extracted from the ROI to ensure all regions have the same size.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 20 November 2020 doi:10.20944/preprints202011.0527.v1

- *Classification Task:* Regions are passed on to a fully connected network which classifies them into different image classes. In our case study, the classes are dent 'Damage' or background 'aircraft skin without damage.'
- *Regression Task:* A bounding box (BB) regression is used to predict the bounding boxes for each identified region for tightening the bounding boxes.

Since aircraft dents don't have a clearly defined shape, arriving at square/rectangular shaped BBs is not sufficient. it's important to identify the exact pixels in the bounding box that corresponds to the class damage. Exact pixel location of the dent will help to identify the location and quantify the damage. An additional step is needed: semantic segmentation (pixel-wise shading of the class of interest) into the entire pipeline for which we will use Masked Region based CNN (Mask R-CNN) architecture.

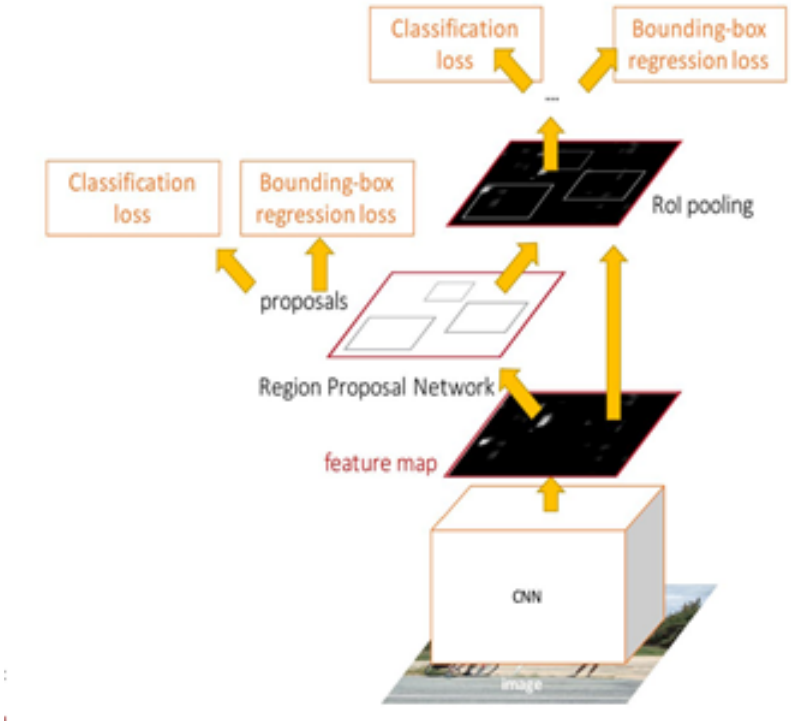


Figure 1. Faster R-CNN architecture based on [28]

2.2. Mask R-CNN

Mask R-CNN is an instance segmentation model, which enables the identification of pixel-wise delineation of the object class of interest. In order to get instance segmentation for a particular image, two main task are required (see also Figure 2):

- *BB based object detection (Localization Task):* uses similar architecture as faster R-CNN. The only difference in Mask R-CNN is the **ROI** step. Instead of using ROI pooling, it uses ROI align to allow the pixel to pixel preserve of ROIs and prevent information loss.
- *Semantic segmentation:* which allows segmenting individual objects at pixel within a scene, irrespective of the shapes. Semantic segmentation uses a **Fully Convolutional** Network (FCN) which creates binary masks around the BB objects through creating pixel-wise classification of each region. Hence, Mask R-CNN minimizes the total loss.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 20 November 2020 doi:10.20944/preprints202011.0527.v1

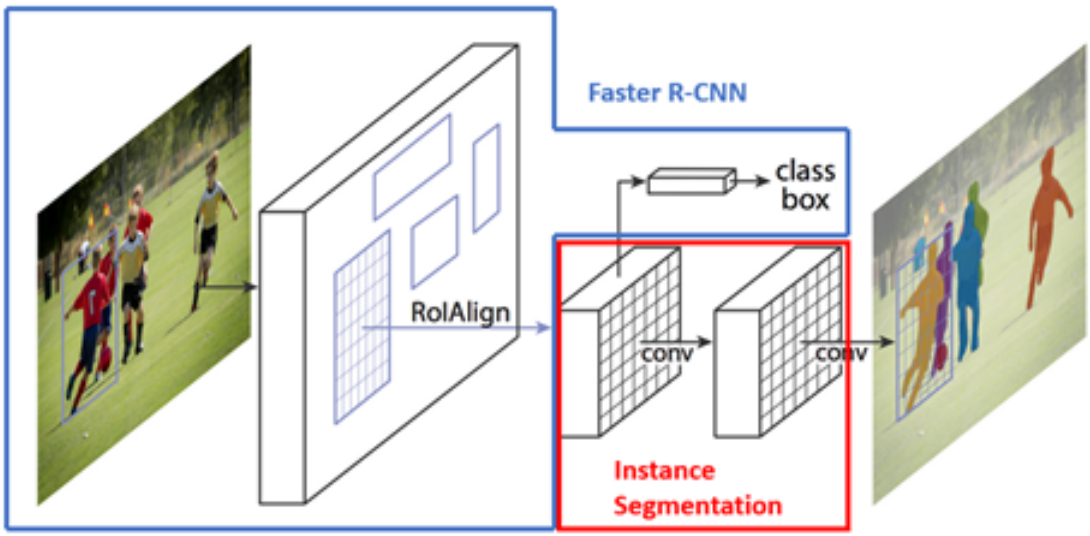


Figure 2. Mask R-CNN framework for instance segmentation [29]

2.3. Implementation

This section discusses the data preparation and the implementation of the concept on real-life aircraft images using Mask R-CNN. The authors have adopted the code take from [27] such that it can be used to identify dents on aircraft structures. In order to reduce the computational time to train the Mask R-CNN, we have applied transfer learning [30] with a warm restart (shown in Figure 3) and taken the initial weights from [31]. By pre-training the neural network on the **COCO**, we then re-use it on our target data set as the lower layers are already trained on recognizing shapes and sizes. In this way we refine the upper layers for our target data set (aircraft structures with dents).

Figure 3. Mask R-CNN framework for instance segmentation

2.4. Environment Set-Up

The most crucial element before training the model is setting up a proper environment, where the core computations are performed. Here we resort to **Google Colab** in combination with **Python, Jupyter notebook**. Google Colab is a free, in-the-browser, collaborative programming environment that provides an interactive and easy to use platform for deep learning researchers and engineers to work on their datascience projects. There is no need for the user to follow complex and tedious procedures to install software, associated packages, worry about data management and computational resources (CPU/GPU/TPU). All is pre-configured and the user can focus directly on the research questions. Google colab is a perfect environment for testing Deep Learning based projects before going

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 20 November 2020 doi:10.20944/preprints202011.0527.v1

into production settings and also provides loads of extras, like documenting your work in Markdown, Version control and Cloning.

3. Experimental Results

This section provides an overview of the performance metrics, experimental set-up, and a summary of the key results.

3.1. Model Performance Evaluation

This section presents the evaluation criteria used to assess model performance. As explained above, Mask R-CNN is used to detect the dents on the given aircraft images (i.e., aircraft defects). From the point of view of the decision makers utilizing such a decision-support system, detecting the dent area is more important than calculating the exact area of the dents accurately. Therefore, this work focuses on accurately detecting the dents and measuring the performance by considering how well the dent predictions are made. For this purpose, the well known prediction performance metrics such as precision, recall and F1 scores are used. In this study, **precision** measures the percentage of truly detected dents among the dent predictions by the given model (i.e, the percentage of detected dents that were correctly classified) while **recall** measures what percentage of the dents predictions that are correctly detected.

Formally, Equation 1 and Equation 2 show how to calculate the precision and recall respectively where:

- **TP**: denotes the true positives and is equal to the number of truly detected dents (i.e., the number of dent predictions, which is correct according to the labeled data).
- **FP**: denotes the false positives and is equal to the number of falsely detected dents (i.e., the number of dent predictions, which are not correct accordingly to the labeled data).
- **FN**: denotes the false negatives and is equal to the number of dents, which are not detected by the model (i.e., the number of dents labelled in the original data but the model could not detect them).

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

In addition to the above metrics, we also consider an extra performance metric, called F_β**-score** (F_β measure). This metric is basically a weighted combination of the Precision and Recall. Besides, the range of the F_β-score is between zero and one where higher values are more desired. In this study, we took two different beta values into consideration which are 1 and 2. F_1 conveys the balance between precision and recall while F_2 weighs recall higher than precision.

$$F_\beta = (1 + \beta^2) * \frac{Precision * Recall}{\beta^2 * Precision + Recall} \tag{3}$$

3.2. Experimental Setup

This section describes the experimental setup and characteristics of datasets used to train and test the convolutional neural network.

3.2.1. Data Collection and Annotation

The first step in this research involves collecting images of aircraft dents from different sources. To the best of the authors knowledge, this is the first study which aims at automatically detecting aircraft dents. Therefore, there was no image database for aircraft dents publicly available. So a key

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 20 November 2020 doi:10.20944/preprints202011.0527.v1

first step was to develop an aircraft dents database from scratch. This was achieved by taking photos of aircraft dents at Abu Dhabi Polytechnic Hangar (4) and combining it with online images that had one or multiple aircraft dents. Since the total number of images was small (56 images), we have involved highly experienced aircraft maintenance engineers during the annotation process in order to accurately label the location of the dents in each image as shown in Figure 5.

Figure 4. Abu Dhabi Polytechnic Hangar

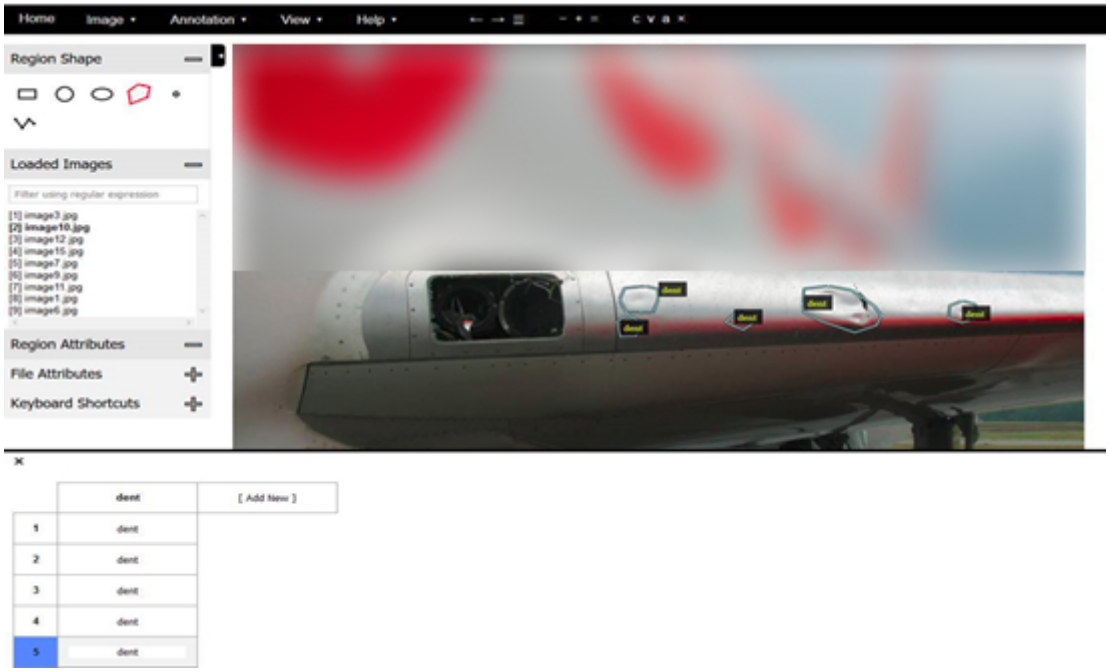


Figure 5. Manual Dent Annotation

3.2.2. Datasets Characteristics

The 56 collected images of aircraft dents were diverse in terms of background color; size & location of dents; causes of dents; resolution; and distance/angle from which the photos were taken. Based on this original dataset we have prepared4 different datasets which are described below and summarized in Table 1.

1. **Dataset 1:** This dataset is a combination of the original dataset which contains 56 images of aircraft dents [1] and a new dataset of 49 images without dents. The annotation in the original dataset used in [1] has also been improved through involving more experts to reach consensus

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 20 November 2020 doi:10.20944/preprints202011.0527.v1

Table 1. Data Set Description

	Image with dents	Images without dents	Scope
Dataset 1	56	49	Aircraft
Dataset 2	26	20	Wing
Dataset 3	56	0	Aircraft
Dataset 4	56	0	Aircraft
Dataset 5	56	49	Aircraft
Dataset 6	56	49	Aircraft

and later verified by another expert. Briefly, Dataset 1 has nearly balanced images with dents and without dents (105 images in total).

2. **Dataset 2:** This dataset is a subset of dataset 1 and contains 46 wing images in total 26 of which have dents, and 20 without dents.
3. **Dataset 3:** This dataset contains half the number of images in the original dataset which contain images with dents only [1], combined with augmented images of the remaining half. Note that we applied the mixed augmentation technique as shown in Figure 7.
4. **Dataset 4:** This dataset contains all the images with dents in the original dataset (56 images with dents) in combination with their augmented version.
5. **Dataset 5:** This dataset contains half the number of images in dataset 1 combined with the augmented images of the remaining half. This dataset contains both images with dents and without dents.
6. **Dataset 6:** This dataset contains all the images with dents in dataset 1 (56 images with dents and 49 images without dents) in combination with their augmented version.

3.2.3. Training and Test Split

The main challenge in this study faced was data scarcity. In addition to using clean and clearly labeled data, we used a 10-fold cross-validation [32] in order to have a diverse pool of training and test data for a robust evaluation. In this approach, the original dataset was split into 10 equally sized parts. By combining these parts in a systematic way (i.e., one for test, the rest for training), we create 10 different combinations of training and test dataset as shown in Figure 6.

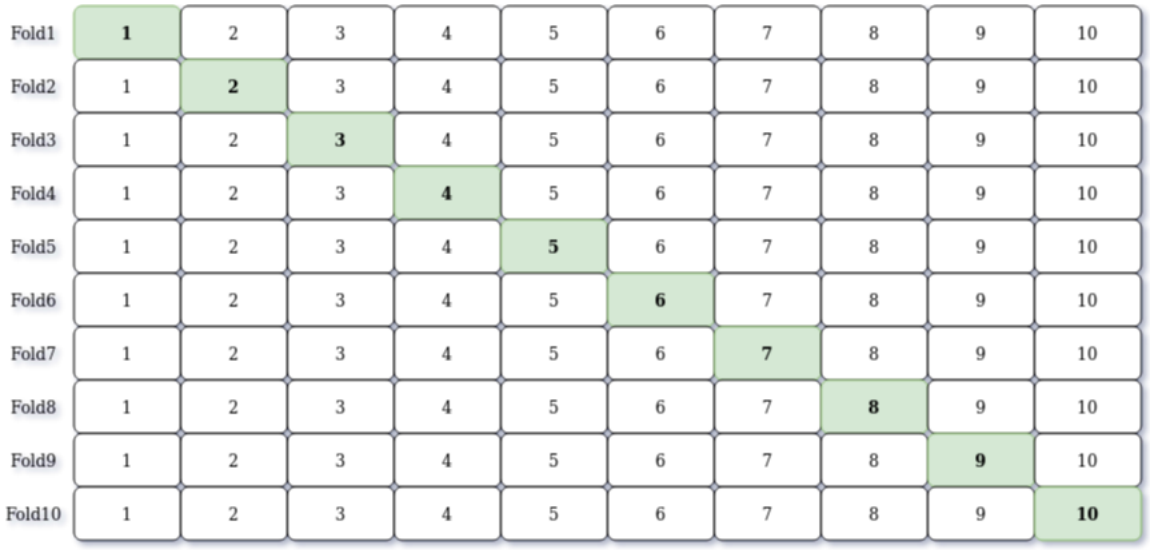


Figure 6. Visualization of 10 Fold Cross Validation
Firstly the dataset is shuffled and then divided into 10 equal pieces. For each fold, one piece is reserved for testing while the remaining ones are used for training. In this figure, the green pieces indicate those reserved for testing while the white ones belong to those used for training. Thus, each fold has different test data.

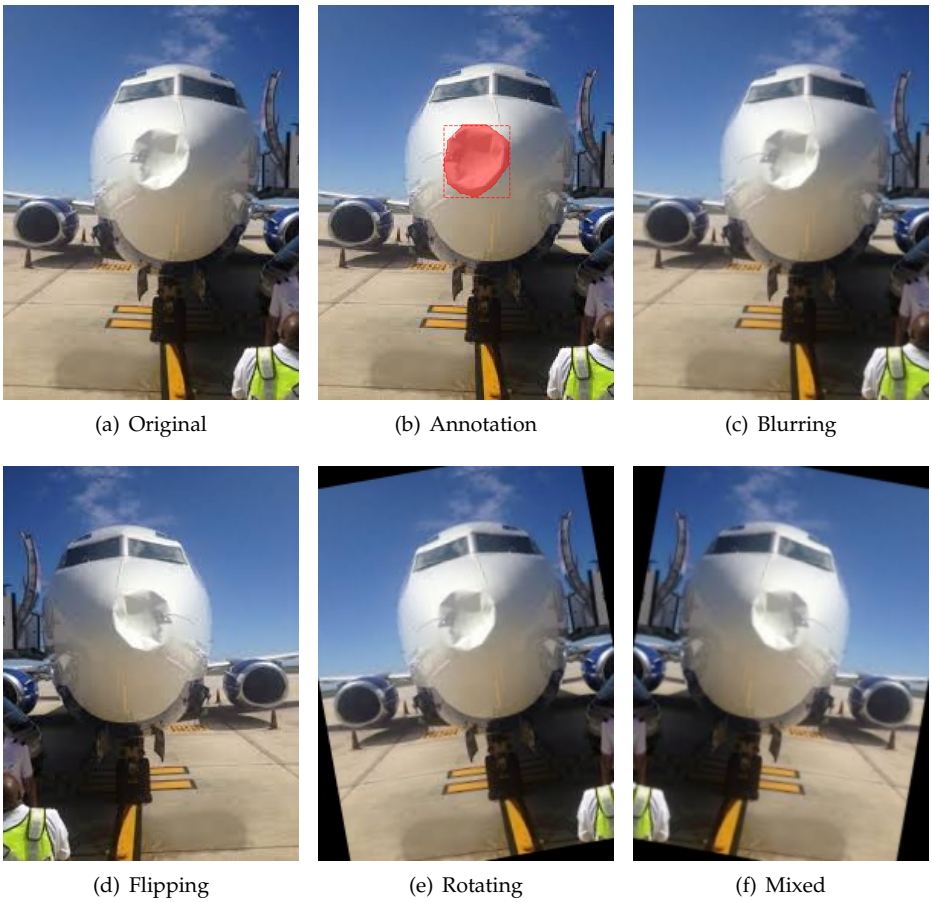
Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 20 November 2020 doi:10.20944/preprints202011.0527.v1

After training the network model on the training set of each fold and testing on the associated test sets separately, an expert checked and compared the predictions with the labeled data for each fold and calculate the true positives TP, false negatives FN, and false positives FP. It is worth noting that we have used a Mask R-CNN which has already been trained to detect car dents [33]. Therefore, even with a small dataset, we could be able to detect the areas of dents on the aircraft dataset. This concept is also known as transfer learning.

3.2.4. Image Augmentation

Image augmentation is a technique which aims at generating new images from already existing ones through a wide range of operations including resizing, flipping, cropping, etc. The purpose of this approach is to create diversity, avoid overfitting, and improve generalizability [34].

In some of the experiments, three augmentation techniques are applied which are flipping, rotating and blurring. This process leads to new images whiles keeping the dents annotations unaffected. Hence, the approach generates new samples with the same label and annotations from already existing ones by visually changing them. In order to prevent damaging the dents images and preserve the image quality, it was decided to use soft augmentation techniques. The techniques were randomly applied to the same image together using a Python library known by imgaug [35]. An example is provided in Figure 7 to illustrate the effects of these techniques.



(a) Original (b) Annotation (c) Blurring

(d) Flipping (e) Rotating (f) Mixed

Figure 7. Image Augmentation Example
The effect of each augmentation technique and the mixed one are shown. As seen in the figures, the dented area is not visually damaged by augmentation techniques.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 20 November 2020 doi:10.20944/preprints202011.0527.v1

3.2.5. Training Approach

Thanks to transfer learning, the ResNet part of the model can extract some visual features that can be utilized in this study without any additional training. However, the other parts of the model must be trained to utilize these visual features. Therefore, the heads of the model (excluding ResNet) must be trained. Firstly, the ResNet weights are frozen, then the model is trained 15 epochs for a dataset of approximately 50 images. Note that the number of epochs are tuned according to the size of the dataset (e.g., 30 for a dataset of 100 images). In addition to this, the ResNet part of the model should be also trained to get better results, because the ResNet may extract more useful visual features after training. Therefore, the weights of the model, including ResNet, is continued training 5 more epochs (also tuned according to the size of the dataset). Briefly, the model is trained for 15 epochs without ResNet, then 5 more epochs with ResNet, a total of 20 epochs is trained.

3.2.6. Pre-Classifier

In Dataset 1, Dataset 5 and Dataset 6, there are some images without dents that the Mask R-CNN model may predict some dents on. This would lead to false positives which would decrease precision. To avoid mispredictions on images without dents, we trained an image classifier to detect whether a given image has dent or not. This approach will significantly increase the precision value. However, it may slightly decrease the recall value when an image with dent is predicted as without dent. For classification, Bag of Visual Words (BoVW) [36] and Support Vector Machine (SVM) [37] are implemented. The average performance results of the pre-classifier model are shown in Table 2.

Table 2. The Performance Results of Classification Model

	Accuracy	Precision	Recall	F1
Training	97.04%	97.0%	97.0%	97.0%
Test	88.82%	89.9%	88.8%	88.7%

For each fold, a pre-classifier was trained on corresponding train set and the metrics are calculated on corresponding test set. In this table, the metrics are the mean of the metrics of all folds.

This approach will be used during testing. First of all, the classifier model generates BoVW vector from a given image in test set. Then, it predicts whether the image is with dent or without dent by classifying the BoWV vector of the image with SVM. The Mask-RCNN model will extract dented areas from the image if it is predicted as image with dent by the classifier. Otherwise, the image will be ignored. The approach is demonstrated in Figure 8.

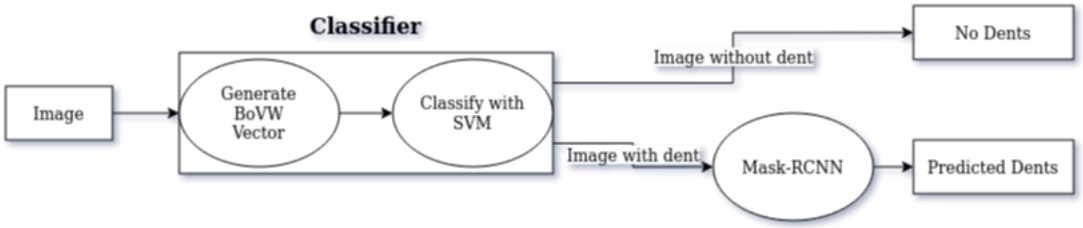


Figure 8. Visualization of The Pre-Classification Approach

3.3. Results & Analysis

This section provides the experimental results showing the prediction performance of the proposed approach.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 20 November 2020 doi:10.20944/preprints202011.0527.v1

Table 3. Overall

Experiment ID	Dataset	Augmentation	Classification	Epoch	Train Size	Test Size	Precision	Recall	F_1 Score	F_2 Score
0	Original Dataset [1]	No	No	15+5	49.5	5.5	69.13%	57.32%	62.67%	59.35%
1	Dataset 1	No	No	30+10	94.5	10.5	38.10%	61.27%	46.98%	54.62%
2	Dataset 1	No	Yes	30+10	94.5	10.5	61.91%	60.68%	61.29%	60.92%
3	Dataset 2	No	No	15+5	41.4	4.6	69.88 %	54.39%	61.17%	56.91%
4	Dataset 3	Yes	No	15+5	50.4	5.6	60.32%	68.08%	63.96%	**66.37%**
5	Dataset 4	Yes	No	30+10	100.8	5.6	**72.48%**	55.01%	62.55%	57.80%
6	Dataset 5	Yes	No	30+10	94.5	10.5	38.85%	**69.97%**	49.96%	60.31%
7	Dataset 5	Yes	Yes	30+10	94.5	10.5	59.17%	68.05	63.30%	66.06%
8	Dataset 6	Yes	No	60+20	189	10.5	44.66%	64.56%	52.80%	59.28%
9	Dataset 6	Yes	Yes	60+20	189	10.5	71.31%	64.08%	**67.50**%	65.41%

3.3.1. High Level Results Overview

The overall table [Table 3] demonstrates the experiments with details and the results. Besides, the measurements are precision, recall, F_1 and F_2 scores. The experiment 0 is from [1]. The experiment 0 is considered as baseline experiment, and the all comparisons are made on this baseline experiment. The highest precision is reached in experiment 5 while the highest recall is reached in experiment 6. In addition, the highest F_1 score is reached in experiment 9, and the highest F_2 score is reached in experiment 4. The details of each experiment are presented in Appendix A and discussed below.

3.3.2. Detailed Results

1. **Experiment 1: Adding images without dents**:

   The main challenge faced was the small size of the dents dataset. To overcome this obstacle, we ensured that the dataset is clean and accurately labeled by involving experienced aircraft engineers. The initial dataset was also extended with new images without dents to improve performance (See Dataset 1). The model is trained 40 epochs in total on Dataset 1. In this experiment, a higher recall value (61.27% versus 57.32%) and lower precision value (38.10% versus 69.13%) have been achieved compared to the baseline experiment conducted in [1]. In this context, recall is more important than precision. Detecting an approximate location of dents correctly is of paramount importance. Our primary aim is not to miss any dents to help human experts analyzing thousands of images. In such a case, it may be admissible if the algorithm may sometimes detect a dent location, which does not exist. In this case, the human expert can give feedback to the system. The detailed results of experiment 1 are shown in Table A1 (Recall: 61.27%; Precision: 38.10%; F_1-Score: 46.98%; F_2-Score: 54.62%).
2. **Experiment 2: Adding images without dents and testing with pre-classifier**

   In experiment 1, considerably lower precision value than the baseline experiment's precision was observed due to high False Positive. The most of False Positive predictions (predicting an area as dent where has no dent) are made on some of the images without dents in dataset 1. Therefore, a classifier (See Section 3.2.6) which predicts whether a given image is with dent or without dent was implemented and used on test set to avoid mispredictions on the images without dent. Firstly, the pre-classifier predicts an image if it has dent, or not. Then, Mask-RCNN model extracts the dented areas if the image is classified as an image with dent. Otherwise, the Mask-RCNN model does not handle the image. In this experiment, we used the Mask-RCNN model trained in experiment 1. The precision value of the experiment dramatically increased from 38.10% to 61.91% by reducing some of False Positive detections. Also, this approach increased not only F_1 score (46.98% to 61.29%) but also F_2 score (54.62% to 60.92%). However, the pre-classifier predicts some of the images with dent as image without dent, so the recall value of the experiment slightly decreased (61.27% to 60.68%). The detailed results of experiment 2 are shown in Table A2 (Recall: 60.68%; Precision: 61.91%; F_1-Score: 61.29%; F_2-Score: 60.92%).

 doi:10.20944/preprints202011.0527.v1

3. **Experiment 3: Filtering the dataset by focusing on only aircraft wings**:

   In machine learning, a specific model with a specific dataset may lead to better results than a generic model. Therefore, a sub-dataset can be prepared by focusing on specific aircraft parts like wing or engine to train a branched model instead of a generic model. Since aircraft dents are often prevalent in areas like the wing leading edge, engines, and radome, this study has focused on the wing because of the data availability. The wing dataset 2 was therefore used to train a branched model that is able to detect wing dents. According to the results, the precision is slightly higher than in the baseline experiment (69.88% versus 69.13%), but the recall (54.39% versus 57.32%), F_1 score (61.17% versus 62.67%) and F_2 score (56.91% versus 59.35%) are slightly lower than the baseline experiment. The results corresponding to experiment 3 are shown in Table A3 (Recall: 54.39%; Precision: 69.88%; F_1-Score: 61.17%; F_2-Score: 56.91%).
4. **Experiment 4: Flipping, rotating, and blurring 50% of dataset**:

   Image augmentation is a technique which aims at generating new images from already existing ones through a wide range of operations including resizing, flipping, cropping, etc. The purpose of this approach is to create diversity, avoid overfitting, and improve generalizability [34]. In experiment 4, half of the images were transformed using three augmentation techniques namely flipping, rotating, and blurring [Section 3.2.4], while the other half remained the same resulting into a new dataset [Dataset 3]. The recall value and F_1 score is higher than the baseline experiment (68.08% versus 57.32% and 63.96% versus 62.67%). Besides, the highest F_2 score among all experiments are obtained in this experiment, although the precision is lower than the baseline experiment (60.32% versus 69.13%). The results of experiment 4 are shown in Table A4 (Recall: 68.08%; Precision: 60.32%; F_1-Score: 63.97%; F_2-Score: 66.37%).
5. **Experiment 5: Flipping, rotating, and blurring the complete dataset**:

   Instead of partially augmenting the dataset as in experiment 4, in this experiment we augment all images and use both original and augmented images for training. Consequently, the dataset [Dataset 4] becomes twice the size of original dataset [Dataset 4] in training phrase. For a fair performance evaluation, we double the number of epoch during the training (15+5 to 30+10). Note that the same image augmentation techniques have been used (flipping, rotating and blurring). Besides, the highest precision among all experiments is reached in this experiment (72.48%). The results corresponding to experiment 5 are shown in Table A5 (Recall: 55.01%; Precision: 72.48%; F_1-Score: 62.55%; F_2-Score: 57.80%).
6. **Experiment 6: Flipping, rotating and blurring 50% of dataset containing images with and without dent**:

   This experiment is combination of the augmentation approach in experiment 4 and adding the images without dent approach in experiment 1. In other words, the same image augmentation approach used in experiment 4 is applied on dataset 5 which contains both 56 images with dent and 49 images without dent. The highest recall value among all experiments is reached in this experiments while the precision is lower than the baseline experiment (38.85% versus 69.13%). The results corresponding to experiment 6 are shown in Table A6 (Recall: 69.97%; Precision: 38.85%; F_1-Score: 49.96%; F_2-Score: 60.31%)
7. **Experiment 7: Flipping, rotating and blurring 50% of dataset containing images with and without dent by testing with the pre-classifier**:

   In the experiment 2, we used the pre-classifier (See Section 3.2.6) to increase the precision value of the experiment 1. Likewise, we used the pre-classifier with the Mask-RCNN model trained in experiment 6 on test set of dataset 5. This approach significantly increases the precision value, F_1 and F_2 scores (38.85% to 59.17%, 49.96% to 63.30% and 60.31% to 66.06%). However, the recall value decreases (69.97% to 68.05%) due to the fact that the pre-classifier predicts some of the images with dent as image without dent. The results corresponding to experiment 7 are shown in Table A7 (Recall: 68.05%; Precision: 59.17%; F_1-Score: 63.30%; F_2-Score: 66.06%).

 doi:10.20944/preprints202011.0527.v1

8. **Experiment 8: Flipping, rotating, and blurring the complete dataset containing images with and without dent**:

   This experiment is combination of the augmentation approach in experiment 5 and adding images without dent approach in experiment 1. In other words, the same image augmentation approach used in experiment 5 is applied on dataset 6 which contains both 56 images with dent and 49 images without dent. Besides, we also double the number of epoch during the training for a fair performance evaluation as in experiment 5. On the other hand, the recall is higher than experiment 5 (64.56% versus 55.01%), and the precision is higher than experiment 1 (44.66% versus 38.10%). Additionally, the recall is also higher than the baseline experiment [1] (64.56% versus 57.32%). The results corresponding to experiment 8 are shown in Table A8 (Recall: 64.56%; Precision: 44.66%; F_1-Score: 52.80%; F_2-Score: 59.28%).
9. **Experiment 9: Flipping, rotating, and blurring the complete dataset containing images with and without dent by testing with the pre-classifier**:

   As in experiment 2 and experiment 7, we used the pre-classifier (See Section 3.2.6) to increase the precision of the experiment 8. In other words, the pre-classifier approach and the Mask-RCNN model trained in experiment 8 are utilized to decrease False Positive detection on the images without dent. The precision considerably increased (44.66% to 71.31%) and the highest F_1 score among all experiments is achieved. Besides, F_2 score increased (59.28% to 65.41%) although the recall value slightly decreased (64.56% to 64.08%) due to misprediction made by the pre-classifier. The results corresponding to experiment 9 are shown in Table A9 (Recall: 64.08%; Precision: 71.31%; F_1-Score: 67.50%; F_2-Score: 65.41%).

4. Conclusion

Aircraft maintenance programs are focused on preventing defects which makes it difficult to collect large datasets of anomalies. Aircraft operators may have 100 images or less for a particular defect. This makes it challenging to develop deep learning aircraft inspection systems based on small datasets. Most of the popular tools are designed to work with big data as used by web companies e.g. using millions of datapoints from users. When the dataset size is limited, it becomes difficult to train the model. To address this problem, we have involved multiple experienced maintenance engineers in annotating the dataset images and then verified the annotation by a third party. That is, we ensured that the dataset is clean and accurately labeled and used augmentation techniques to overcome the small data obstacle.

To train the model, we used MASK R-CNN in combination with augmentation techniques. The model was trained with different datasets to better understand the effect on performance. In total, nine experiments were conducted and performance was evaluated using four metrics namely Precision, Recall, F1 and F2 scores. The experiment variables included the number of epochs, augmentation approaches, and the use of an image pre-classifier. Overall, the highest F1 score (67.50%) corresponds to experiment 9, and the highest F2 score (66.37%) corresponds to experiment 4. Experiment 4 used augmentation techniques such as flipping, rotating, and blurring but only on half of the dataset, while in Experiment 9 all images with and without dents have been augmented. In addition, a pre-classifier was used to prevent mispredictions on images without dents in Experiment 9 (see figure 8). According to our results, it seems that using a pre-classifier improved the prediction performance especially in terms of F1 score. Moreover, it can be concluded that for such a small data problem, a hybrid approach which combines MASK R-CNN and augmentation techniques leads to improved performance.

Future work should be geared towards exploring the effects of various architectures on the performance of detecting aircraft dents. Since MASK R-CNN consists of the RESNET and FPN layers, it would be interesting to investigate other architectures such as U-net. Furthermore, since this study only explored three augmentation techniques, one can investigate additional techniques such as resizing, shear, elastic distorions, and lighting. Another important line of research is AI deployment. Developing a deep learning visual inspection system can be completed by conducting

 doi:10.20944/preprints202011.0527.v1

offline experiments under a highly controlled environment; however, getting to a deployable solution in an MRO environment ready and then scaling it is a long way to go [38] . There need to be more experiments to overcome a complex set of obstacles including the ability to detect defects under varying conditions (e.g. diurnal and environmental effects) and dealing with various uncertain variables.

Appendix A

Table A1. The Results of Experiment 1: Adding images without dent

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Average
Train	94	94	94	94	94	95	95	95	95	95	94.5
Test	11	11	11	11	11	10	10	10	10	10	10.5
TP	3	5	5	59	6	23	5	8	3	4	12.1
FP	14	21	12	6	8	19	13	5	22	12	13.2
FN	5	5	3	81	0	56	2	2	2	1	15.7
Recall	37.50%	50.00%	62.50%	42.14%	100.00%	29.11%	71.43%	80.00%	60.00%	80.00%	**61.27%**
Precision	17.65%	19.23%	29.41%	90.77%	42.86%	54.76%	27.78%	61.54%	12.00%	25.00%	**38.10%**

Table A2. The Results of Experiment 2: Adding images without dents and testing with pre-classifier

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Average
Train	94	94	94	94	94	95	95	95	95	95	94.5
Test	11	11	11	11	11	10	10	10	10	10	10.5
TP	3	5	5	54	6	23	5	8	3	4	11.6
FP	8	2	6	4	3	3	5	4	7	1	4.3
FN	5	5	3	95	0	56	2	2	2	1	17.1
Recall	37.50%	50.00%	62.50%	36.24%	100.00%	29.11%	71.43%	80.00%	60.00%	80.00%	**60.68%**
Precision	27.27%	71.43%	45.45%	93.10%	66.67%	88.46%	50.00%	66.67%	30.00%	80.00%	**61.91%**

Table A3. The Results of Experiment 3: Filtering the dataset by focusing on only aircraft wings

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Average
Train Size	41	41	41	41	41	41	42	42	42	42	41.4
Test Size	5	5	5	5	5	5	4	4	4	4	4.6
TP	2	3	5	6	15	1	1	1	9	1	4.4
FP	2	0	2	1	5	1	5	0	0	1	1.7
FN	1	2	1	2	12	2	3	1	11	1	3.6
Recall	66.7%	60.0%	83.3%	75.0%	55.6%	33.3%	25.0%	50.0%	45.0%	50.0%	**54.39%**
Precision	50.0%	100.0%	71.4%	85.7%	75.0%	50.0%	16.7%	100.0%	100.0%	50.0%	**69.88%**

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 20 November 2020 doi:10.20944/preprints202011.0527.v1

Table A4. The Results of Experiment 4: Flipping, rotating and blurring 50% of dataset

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Average
Train Size	50	50	50	50	50	50	51	51	51	51	50.4
Test Size	6	6	6	6	6	6	5	5	5	5	5.6
TP	34	8	5	22	5	9	5	4	25	27	14.4
FP	2	12	5	13	5	4	2	16	18	4	8.1
FN	26	2	4	3	1	4	0	1	52	49	14.2
Recall	56.7%	80.0%	55.6%	88.0%	83.3%	69.2%	100.0%	80.0%	32.5%	35.5%	**68.08%**
Precision	94.4%	40.0%	50.0%	62.9%	50.0%	69.2%	71.4%	20.0%	58.1%	87.1%	**60.32%**

Table A5. The Results of Experiment 5: Flipping, rotating and blurring the complete dataset

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Average
Train	50	50	50	50	50	50	51	51	51	51	50.4
Test	6	6	6	6	6	6	5	5	5	5	5.6
TP	13	6	6	17	5	9	4	2	20	30	11.2
FP	1	3	6	4	2	4	1	3	6	1	3.1
FN	45	4	3	8	1	5	1	3	57	46	17.3
Recall	22.41%	60.00%	66.67%	68.00%	83.33%	64.29%	80.00%	40.00%	25.97%	39.47%	**55.01%**
Precision	92.86%	66.67%	50.00%	80.95%	71.43%	69.23%	80.00%	40.00%	76.92%	96.77%	**72.48%**

Table A6. The Results of Experiment 6: Flipping, rotating and blurring 50% of dataset containing images with and without dent

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Average
Train	94	94	94	94	94	95	95	95	95	95	94.5
Test	11	11	11	11	11	10	10	10	10	10	10.5
TP	4	8	6	72	6	23	7	8	3	4	14.1
FP	17	18	11	13	13	13	17	8	9	17	13.6
FN	4	2	2	86	0	56	0	2	2	1	15.5
Recall	50.00%	80.00%	75.00%	45.57%	100.00%	29.11%	100.00%	80.00%	60.00%	80.00%	**69.97%**
Precision	19.05%	30.77%	35.29%	84.71%	31.58%	63.89%	29.17%	50.00%	25.00%	19.05%	**38.85%**

Table A7. The Results of Experiment 7: Flipping, rotating and blurring 50% of dataset containing images with and without dent by testing with the pre-classifier

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Average
Train	94	94	94	94	94	95	95	95	95	95	94.5
Test	11	11	11	11	11	10	10	10	10	10	10.5
TP	4	8	6	39	6	23	7	8	3	4	10.8
FP	11	7	7	6	6	3	9	2	3	2	5.6
FN	4	2	2	109	0	56	0	2	2	1	17.8
Recall	50.00%	80.00%	75.00%	26.35%	100.00%	29.11%	100.00%	80.00%	60.00%	80.00%	**68.05%**
Precision	26.67%	53.33%	46.15%	86.67%	50.00%	88.46%	43.75%	80.00%	50.00%	66.67%	**59.17%**

 doi:10.20944/preprints202011.0527.v1

Table A8. The Results of Experiment 8: Flipping, rotating, and blurring the complete dataset containing images with and without dent

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Average
Train	188	188	188	188	188	190	190	190	190	190	189
Test	11	11	11	11	11	10	10	10	10	10	10.5
TP	4	7	6	47	5	26	6	8	3	3	11.5
FP	11	14	10	7	17	8	14	12	3	4	10
FN	4	3	2	100	0	53	1	2	2	2	16.9
Recall	50.00%	70.00%	75.00%	31.97%	100.00%	32.91%	85.71%	80.00%	60.00%	60.00%	**64.56%**
Precision	26.67%	33.33%	37.50%	87.04%	22.73%	76.47%	30.00%	40.00%	50.00%	42.86%	**44.66%**

Table A9. The Results of Experiment 9: Flipping, rotating, and blurring the complete dataset containing images with and without dent by testing with the pre-classifier

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Average
Train	188	188	188	188	188	190	190	190	190	190	189
Test	11	11	11	11	11	10	10	10	10	10	10.5
TP	4	7	6	40	5	26	6	8	3	3	10.8
FP	7	5	3	3	3	0	5	4	0	1	3.1
FN	4	3	2	107	0	53	1	2	2	2	17.6
Recall	50.00%	70.00%	75.00%	27.21%	100.00%	32.91%	85.71%	80.00%	60.00%	60.00%	**64.08%**
Precision	36.36%	58.33%	66.67%	93.02%	62.50%	100.00%	54.55%	66.67%	100.00%	75.00%	**71.31%**

References

1. Bouarfa, S.; Doğru, A.; Arizar, R.; Aydoğan, R.; Serafico, J. Towards Automated Aircraft Maintenance Inspection. A use case of detecting aircraft dents using Mask R-CNN. AIAA Scitech 2020 Forum, 2020, p. 0389.
2. Drone, M. MRO Drone: RAPID. https://www.mrodrone.net/, (accessed: 22.09.2020).
3. mainblades. mainblades: Aircraft lightning strike inspection. https://mainblades.com/lightning-strike-inspection/, (accessed: 22.09.2020).
4. Boeing. Pilot & Technician Outlook 2019-2038. https://www.boeing.com/commercial/market/pilot-technician-outlook/, (accessed: 22.09.2020).
5. Aeronews. ATR72 Missed Damage: Maintenance Lessons. http://aerossurance.com/safety-management/atr72-missed-damage/, (accessed: 25.09.2020).
6. Aeronews. Google Brain chief: AI tops humans in computer vision, and healthcare will never be the same. https://siliconangle.com/2017/09/27/google-brain-chief-jeff-dean-ai-beats-humans-computer-vision-healthcare-will-never/, (accessed: 25.09.2020).
7. Esteva, A.; Robicquet, A.; Ramsundar, B.; Kuleshov, V.; DePristo, M.; Chou, K.; Cui, C.; Corrado, G.; Thrun, S.; Dean, J. A guide to deep learning in healthcare. *Nature Medicine* **2019**, *25*, 24–29. doi:10.1038/s41591-018-0316-z.
8. Esteva, A.; Kuprel, B.; Novoa, R.A.; Ko, J.; Swetter, S.M.; Blau, H.M.; Thrun, S. Dermatologist-level classification of skin cancer with deep neural networks. *nature* **2017**, *542*, 115–118.
9. Cheng, J.Z.; Ni, D.; Chou, Y.H.; Qin, J.; Tiu, C.M.; Chang, Y.C.; Huang, C.S.; Shen, D.; Chen, C.M. Computer-aided diagnosis with deep learning architecture: applications to breast lesions in US images and pulmonary nodules in CT scans. *Scientific reports* **2016**, *6*, 1–13.
10. Gulshan, V.; Peng, L.; Coram, M.; Stumpe, M.C.; Wu, D.; Narayanaswamy, A.; Venugopalan, S.; Widner, K.; Madams, T.; Cuadros, J.; others. Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *Jama* **2016**, *316*, 2402–2410.

11. Cireşan, D.C.; Giusti, A.; Gambardella, L.M.; Schmidhuber, J. Mitosis detection in breast cancer histology images with deep neural networks. International conference on medical image computing and computer-assisted intervention. Springer, 2013, pp. 411–418.
12. Benjamens, S.; Dhunnoo, P.; Meskó, B. The state of artificial intelligence-based FDA-approved medical devices and algorithms: an online database. *npj Digital Medicine* **2020**, *3*, 1–8.
13. Patrício, D.I.; Rieder, R. Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review. *Computers and electronics in agriculture* **2018**, *153*, 69–81.
14. Tian, H.; Wang, T.; Liu, Y.; Qiao, X.; Li, Y. Computer vision technology in agricultural automation—A review. *Information Processing in Agriculture* **2020**, *7*, 1–19.
15. Ganesh, P.; Volle, K.; Burks, T.; Mehta, S. Deep orange: Mask R-CNN based orange detection and segmentation. *IFAC-PapersOnLine* **2019**, *52*, 70–75.
16. Yun, J.P.; Shin, W.C.; Koo, G.; Kim, M.S.; Lee, C.; Lee, S.J. Automated defect inspection system for metal surfaces based on deep learning and data augmentation. *Journal of Manufacturing Systems* **2020**, *55*, 317–324.
17. Ren, R.; Hung, T.; Tan, K.C. A generic deep-learning-based approach for automated surface inspection. *IEEE transactions on cybernetics* **2017**, *48*, 929–940.
18. Weimer, D.; Scholz-Reiter, B.; Shpitalni, M. Design of deep convolutional neural network architectures for automated feature extraction in industrial inspection. *CIRP Annals* **2016**, *65*, 417–420.
19. Luckow, A.; Cook, M.; Ashcraft, N.; Weill, E.; Djerekarov, E.; Vorster, B. Deep learning in the automotive industry: Applications and tools. 2016 IEEE International Conference on Big Data (Big Data). IEEE, 2016, pp. 3759–3768.
20. Paolanti, M.; Romeo, L.; Martini, M.; Mancini, A.; Frontoni, E.; Zingaretti, P. Robotic retail surveying by deep learning visual and textual data. *Robotics and Autonomous Systems* **2019**, *118*, 179–188.
21. Faghih-Roohi, S.; Hajizadeh, S.; Núñez, A.; Babuska, R.; De Schutter, B. Deep convolutional neural networks for detection of rail surface defects. 2016 International joint conference on neural networks (IJCNN). IEEE, 2016, pp. 2584–2589.
22. Malekzadeh, T.; Abdollahzadeh, M.; Nejati, H.; Cheung, N.M. Aircraft fuselage defect detection using deep neural networks. *arXiv preprint arXiv:1712.09213* **2017**.
23. Miranda, J.; Larnier, S.; Herbulot, A.; Devy, M. UAV-based inspection of airplane exterior screws with computer vision. 14h International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications., 2019.
24. Miranda, J.; Veith, J.; Larnier, S.; Herbulot, A.; Devy, M. Machine learning approaches for defect classification on aircraft fuselage images aquired by an UAV. Fourteenth International Conference on Quality Control by Artificial Vision. International Society for Optics and Photonics, 2019, Vol. 11172, p. 1117208.
25. Miranda, J.; Veith, J.; Larnier, S.; Herbulot, A.; Devy, M. Hybridization of deep and prototypical neural network for rare defect classification on aircraft fuselage images acquired by an unmanned aerial vehicle. *Journal of Electronic Imaging* **2020**, *29*, 041010.
26. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. YouOnlyLookOnce: Unified Real-Time Object Detection. https://arxiv.org/pdf/1506.02640v5.pdf, 2016.
27. CNN Application-Detecting Car Exterior Damage(full implementable code). https://towardsdatascience.com/cnn-application-detecting-car-exterior-damage-full-implementable-code-1b205e3cb48c.
28. Shaoqing, R.; Kaiming, H.; Ross, G.; Jian, S. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. https://arxiv.org/pdf/1506.01497.pdf, 2016.
29. He, K.; Gkioxari, G.; Dollar, P.; Girshick, R. Mask R-CNN. https://arxiv.org/pdf/1703.06870.pdf, 2018.
30. Yang.; Pan. A survey on Transfer Learning. https://doi.org/10.1109/TKDE.2009.191, 2010.
31. Github. Releases Mask R-CNN COCO weights h5 file. https://github.com/matterport/Mask_RCNN/releases/download/v2.0/mask_rcnn_coco.h5, 2019.
32. Alpaydın, E. *Introduction to Machine Learning, Fourth Edition*, 4 ed.; MIT Press, 2020.
33. Dey, S. Car damage detection using CNN. https://github.com/nitsourish/car-damage-detection-using-CNN, (accessed: 08.11.2020).
34. Agarwal, S.; Terrail, J.O.D.; Jurie. Recent advances in object detection in the age of deep convolutional neural networks. https://hal.archives-ouvertes.fr/hal-01869779v2/document. Online; accessed 23 October 2020.
35. Jung, A.B. imgaug. https://github.com/aleju/imgaug, 2018. [Online; accessed 30-Oct-2018].

36. Fei-Fei, L.; Fergus, R.; Torralba, A. Recognizing and Learning Object Categories. http://people.csail.mit.edu/torralba/shortCourseRLOC/, 2009. Online.
37. Cortes, C.; Vapnik, V. Support-Vector Networks. *Machine Learning* **1995**, *20*, 273–297.
38. LandingAI. Redefining quality control with AI-powered visual inspection for manufacturing. https://landing.ai/wp-content/uploads/2020/04/LandingAI_WhitePaper_v2.0_FINAL.pdf. Online; accessed 23 October 2020.