Data-centric estimation of wind turbine height using aerial imagery

by

Chris Lill

Data Science MSc

Submitted to the Department of Computing

University of London

March 2023

Abstract

Data-centric AI is the systematic engineering of data to improve performance, and is a mindset that is actively promoted in the data science community. This project measures the benefit of different data-centric techniques. It focuses on a novel task to estimate wind turbine hub height from aerial imagery in Spain, using computer vision models to measure the turbine shadow and the image timestamp to calculate the turbine height.

This project demonstrates that different data-centric techniques are optimal for each use cases, even when using the same images. Effective data-centric tooling is needed, not only in Machine Learning Operations (ML Ops) platforms but also embedded in computer vision models. Computer vision frameworks such as YOLOv7 contain many data-centric transformations in their training code by default, and this contributes significantly to their performance.

We demonstrate that good results can be gained from a small dataset of 76 wind farms, with 70% of wind farms having an hub height error of less than 5m. To achieve these results, additional functionality was required to account for elevation differences between the base and hub shadow, and to ensure the correct timestamp was being used for each turbine.

Table of Contents

List of Figures

List of Tables

1. Introduction

Data-centric AI is “the discipline of systematically engineering the data used to build an AI system” (*Data-centric AI Resource Hub*, 2022). Every AI system is built from a combination of code and data, and data-centric AI proposes that activities to improve the quality and consistency of the data usually have the greatest impact on model performance. This can be compared to model-centric activities such as experimenting with different algorithms, and tuning model hyperparameters and topologies. Model-centric improvements are usually easier for a data scientist to make and have historically received greater focus.

The aim of this project is to measure the benefit from different data-centric techniques for two computer vision models. Specifically, this research will focus on the estimation of the height of wind turbines in Spain from aerial photography, which is an novel application of computer vision in the wind energy industry. Data-centric techniques have often been studied for image classification, but it is less common to study them for object detection where there is arguably greater benefit because of the higher cost of labelling data.

When deciding whether to develop or invest in a wind farm, it is important to reduce the uncertainty of wind conditions (Schneider *et al.*, 2022). One approach is to infer the average wind speed from publicly available power generation data at nearby wind farms. However, this calculation requires knowledge of the hub height, which is not always available on the internet. Wind speed varies significantly according to the height above the ground, so a consistent way of estimating wind turbine hub height could reduce the uncertainty in wind farm valuations.

Wind turbine hub height can be estimated from aerial imagery. A suitable data source for Spain is the Centro de Descargas (Geográfica, 2022). The timestamp of the photo can be

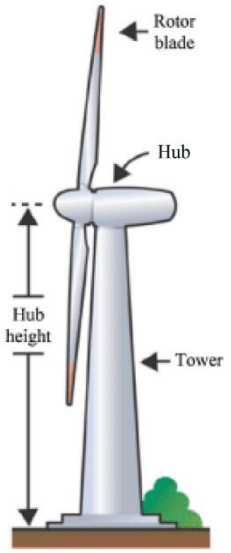

Figure 1: Wind turbine side view. After Dimitrova et al (2022)

Figure 2: Predicted hub shadows for the Paxareiras-Montevós wind farm

1. Turbine shadow 2. Hub shadow 3. Hub height estimation

Figure 3: Three objectives for this project

used to calculate the angle of the sun above the horizon at the time the photo was taken. Trigonometry can then be used to estimate the hub height from the shadow length.

This project measures the length of wind turbine shadows using two computer vision models. The first detects turbine shadows on large aerial photos at lower resolution. The second precisely locates the base and shadow of the hub. This will provide two opportunities to compare the impact of data-centric activities on computer vision performance. These models are then be used on a test dataset where hub heights are known to prove whether this method can be used to estimate the hub height within 5m.

The code to implement this solution is shared on GitHub[1]. Two new datasets have been extracted from aerial imagery, labelled and published on Roboflow[2].

1.1 Objectives and research questions

There are three objectives.

1. Measure the importance of data-centric techniques in identifying the location of wind turbine shadows within a known wind farm. This is the “turbine shadow” task.
2. Measure the importance of data-centric techniques in identifying the location of the hub shadow and turbine base in an image of a wind turbine. This is the “hub shadow” task.
3. To estimate the hub height and measure the accuracy against known hub heights.

[1] https://github.com/chrislill/wind_turbine_height
[2] https://universe.roboflow.com/windturbineheight

This project has three research questions (RQ), one of which can be expressed as a hypothesis.

RQ1. What is the impact of data-centric activities on the turbine shadow task?
RQ2. What is the impact of data-centric activities on the shadow length task?
RQ3. Hypothesis: The average error of the estimated hub height is less than 5m.

The threshold of 5m for research question 3 has been selected because it should enable this approach to be used in real world applications. The hypothesis will be tested with a significance level of 5%.

2. Related work

2.1 Data-centric AI

Andrew Ng proposes that "in this decade… the biggest shift in AI might be a shift to data-centric AI". Every AI system is a combination of code and data. Traditionally AI researchers and practitioners have focused on improvements to the code (model-centric), however systematically improving the data (data-centric) is likely to have greater benefits. Data scientists commonly joke that they spend 80% of their time preparing data, but it is still common practice to hold the data fixed whilst experimenting with different code. Instead, data-centric AI suggests that once we have a working model we hold that fixed and systematically improve the data (Ng, 2021).

There is a heavy bias in academic research and industry for model-centric approaches. Many industry conferences traditionally required a "machine learning contribution" that focuses on the second row in figure 3 below (Wagstaff, 2012). This is at the expense of the data-centric tasks in the first row of the diagram, or the activities in the third row that are required for Machine Learning (ML) to have an impact in the real world. Anecdotal evidence suggests that more than 90% of recent papers are model-centric rather than data-centric (Ng, 2021).

Neglecting data quality does not only impact model performance, it can also lead to project failure. A survey of 53 AI practitioners found that 92% of projects suffered from negative downstream effects triggered by AI/ML practices that undervalue data quality (Sambasivan *et al.*, 2021). Work to prepare and improve data is less interesting and less rewarding. It is seen as time-consuming and is often delegated by data scientists to junior colleagues or subject matter experts. If the quality of data is not improved systematically, the probability that multiple data issues impact the project significantly increases.

Psychologically, humans may have a natural bias towards model-centric approaches. When solving a problem, it is intuitive to take an action on an object, observe the result, and then modify the action we take. Only once the actions become difficult would one think to change the object. Applying this concept to machine learning we can think of our the model

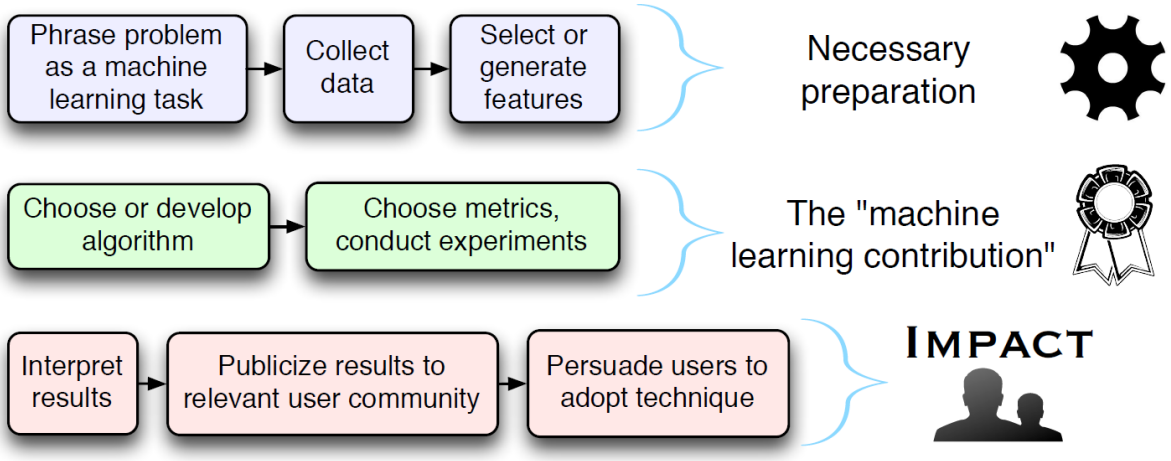

Figure 4: Three stages of a machine learning research programme. Traditionally publishing incentives are highly biased to the second row (Wagstaff, 2012)

code as the action and the data as the object. Data scientists are likely to focus on modifying their code (actions) before changing the properties of the data (object) (Hamid, 2022). Model-centric choices are clearly documented in the code that has been written, which leads to an availability bias where data scientists have a greater tendency to experiment with these settings because they are readily available.

Hamid concludes that model-centric and data-centric approaches are complimentary (2022). The data scientist needs to be aware of both and focus on the tasks that will bring the most benefit. This is implicit, rather than explicitly stated in the original data-centric AI video, and is an important clarification.

Often the data-centric activities that data scientists take are ad-hoc in response to observations or visible errors in the results. This random approach is less effective, leaving smaller errors unaddressed. An important element of a data-centric approach is that it is systematic. Ng (2021) suggests that Machine Learning Operations (ML Ops) is a discipline that should evolve to include “a set of tools and best practices to make data-centric AI efficient and systematic”.

The difficulty and cost of labelling datasets is a key driver towards data-centric techniques. Many data science problems in industry are restricted to smaller labelled datasets, either due to the cost of labelling or because they have a long tail of rare events. Ng (2021) suggests that ‘with a clean consistent dataset it is possible to get good results with 40 - 100 labels’. Moreover he shares an example of 500 labels with 12% noisy data, where the same results could be achieved by cleaning the data or by tripling the size of the dataset. Cleaning the data is likely to be a much easier task. This shows the importance of techniques that can systematically improve the quality of data.

2.2 Computer vision models

A typical technique for measuring distances in images is to use object detection to position a bounding box at either end of the object (Rosebrock, 2016). Object detection models locate objects within an image and classify the type of each object. Traditional models had a different approach for these two tasks, with both needing to be completed during each training epoch. These are known as two-stage detectors. Modern examples of this are the Region-based Convolutional Neural Network (R-CNN) family of models, such as Faster R-CNN (Ren *et al.*, 2015).

You Only Look Once (YOLO) is currently the most popular family of models. They were created by Joseph Redmon (2016) and are single stage object detectors that use a convolutional neural network to predict bounding boxes and class probabilities across the whole image at once. They are fast, generalizable, and consider contextual information from the whole image. However, they can struggle to identify the optimal bounding box and much recent research has focused on improving performance using different loss functions (Sun *et al.*, 2020).

Since Redmon stopped computer vision research due to ethical concerns around face recognition and military uses (Synced, 2020) there have been several rival branches of YOLO development. One branch of development from original YOLO contributors produced YOLOv4 (Bochkovskiy *et al.*, 2020) and YOLOv7 (Wang *et al.*, 2022). These have collected and automated many individual techniques for improving model performance into a "bag of freebies"; these are techniques that improve the performance of the model during training but do not increase the time needed for inference. This includes several data transformations for training images such as colour modification, adding noise, flipping, re-scaling, rotation and shear. This also introduced two new transformations that combine training images: CutMix to improve how the model handles occlusion, and Mosaic to encourage the model to localize different images in different parts of the frame.

These data augmentation techniques are fully integrated into the YOLOv7 model training code and are easily tuned as part of the model configuration. This task can be described as data hyperparameter tuning. Hyperparameters for colour, translate, scale, flip, and mosaic are even included as the default settings. All parameters can be tuned using the "evolve" option which automates hyperparameter tuning using a genetic algorithm. This addresses the availability bias against data-centric techniques in a different way than ML Ops, making it easy to improve the data within the model itself.

Data-centric optimizations can also be made before calling the model. Roboflow is an example of an ML Ops platform that streamlines the labelling process and data pipeline and is well aligned with the concept of data-centric AI. It includes many data-centric transformations to improve computer vision datasets such as rotation, hue, mosaic and

adding noise. It can also generate a model that can be used to suggest initial labels for new images, thus accelerating the labelling process.

2.3 Data centric techniques

2.3.1 Adding to the dataset

Adding new labelled data to the dataset is a common way to improve the performance of the model. Machine learning models perform better when the data they are built with is fully representative of the domain. As data is added model performance often increases according to an inverse power law (Figueroa *et al.*, 2012), so eventually the benefits of more data reduce. However, computer vision problems are usually high-dimension low-sample-size problems where the size of the labelled data is far smaller than the number of parameters in the model (Keshari *et al.*, 2018).

Active learning has been shown to improve efficiency when there is a cost associated with labelling a dataset. Its goal is to identify the additional data that is most likely to improve the model. There are many different strategies for selecting which data to label, depending on the availability of data and query strategy (Settles, 2009). A simplistic approach is uncertainty sampling, where new data is processed by an initial model and the labels with the lowest predicted probabilities are labelled and added to the training set (Lewis and Gale, 1994).

2.3.2 Pre-processing

This paper uses the term data-centric transformations to describe functions that can be applied to each image in the dataset to increase the performance of the model. These can be split into two categories. Pre-processing transformations can be applied to all images, including the test set, to simplify the task. Augmentations are transformations that are just applied to the training and validation data sets to introduce variation in the training data and increase the dataset size.

Pre-processing is important to improve small object detection for aerial imagery. Aerial images cover large areas, and naturally contain many small objects. Using the full resolution of the images could help, but this is impractical because of the computing resource and training time that would be required. Images can be scaled down, but the performance of all modern object detection models reduces at lower resolutions (Wang *et al.*, 2022). Tiling is a pre-processing transformation that can split the image into smaller images that are easier for the model to process. Images can then be scaled to a consistent resolution that makes the right trade-off between resolution and processing power for the dataset.

Contrast pre-processing is a common technique in astronomy where the image can be transformed so that it uses the full range of colour shades, which can help the detection of edges in images. Advanced techniques include histogram equalization where the intensity of each pixel is modified so that the contrast is distributed evenly across the full range of

values. Adaptive histogram equalization is a similar technique operating locally by rescaling each pixel based on the values in a larger square surrounding it (Pizer *et al.*, 1987).

2.3.3 Data augmentation

Augmentations increase the size and diversity of the dataset by adding modified versions of each training and validation image. Their purpose is to create real-looking fake data and to help the model to generalize to images it has not seen. Augmentations can be implemented in an ML Ops platform, or within the model code. The benefit of embedding augmentations in the model code is that a different image can be used for each epoch of training, further increasing the size and diversity of the dataset. However this does require additional training time and resource.

Simple data augmentations include flip, where the image can be reflected in the horizontal or vertical plane. Rotation and shear are augmentations that improve the model's ability to recognise objects at different orientations. Translate moves the image within the frame, helping the model to detect objects in different areas of the image. Colour augmentations can modify the hue, saturation or value of each pixel to generate images with a wider range of lighting conditions or colours.

Adding noise to an image can help the model generalize. A similar concept applied within the model is the dropout layer used in many neural networks. This regularizes a model by dropping a random sample of weights for each training epoch. YOLOv7 uses a similar transformation to dropout called DropBlock, where units in a contiguous region of a feature map are dropped together. DropBlock is more performant than dropout in convolutional networks because it addresses an issue where the spatial correlation of layers reduces regularizing effect (Ghiasi *et al.*, 2018). The equivalent data augmentation that applies noise in contiguous regions in YOLOv7 is paste-in.

Paste-in is an improvement to the established Cutout technique which randomly masks out square regions of input images during training. Rather than replace it with null data, paste-in replaces the masked area with a different image and updates the labels accordingly. This improves the model's ability to detect objects in partially occluded scenes. It does this by encouraging "the network to better utilize the full context of the image, rather than relying on the presence of a small set of specific visual features" (DeVries and Taylor, 2017).

Some data augmentation transformations that are suitable for image classification may introduce inaccuracy when used for object detection. YOLOv7 requires rectangular bounding boxes and if a rotation or shear transformation is applied then the new bounding box might not be consistent with the distorted image. An alternative approach is to only apply these transformations within the bounding box for each label (Zoph *et al.*, 2019). This family of techniques are known as bounding-box transformations and are supported in Roboflow.

2.3.4 Selection of data-centric techniques

Although it is tempting to apply all available data cleaning and transformations, this can introduce bias and reduce the information available to the model. The effect of each data cleaning techniques is very specific to the dataset, being dependent on the variation and noise in the data. Therefore selection of the most appropriate cleaning algorithms is always required (Li *et al.*, 2021).

2.4 Metrics

There are two similar metrics commonly used to evaluate object detection models. The PASCAL Visual Object Competition (VOC) uses average precision, also known as mean Average Precision (mAP_0.5). This metric compares actual and estimated bounding boxes, deeming those with an Intersection over Union (IoU) greater or equal to 0.5 as true positives. The precision/recall curve can then be used to calculate the average precision for recall values between 0 and 1 (Everingham *et al.*, 2010).

The Microsoft Common Objects in Context (COCO) dataset uses a similar metric, mAP_0.5-0.95. This calculates the mAP for ten thresholds of IoU from 0.5 to 0.95 (*COCO - Common Objects in Context*, 2023). This rewards models that achieve higher values of IoU and are better at the precise location of objects. This metric is well suited to object detection tasks, and especially measuring the distance between the centres of two objects.

2.5 Information required to estimate wind speed

Site specific wind measurements are the most accurate way to estimate the potential yield of a wind farm. They usually require the installation of a meteorological mast or Light Detection and Ranging (LIDAR) equipment. Unfortunately this is time consuming and expensive, requiring at least six months of quality data (Tordoff, 2013). For initial estimates, climate reanalysis atlases, such as the New European Wind Atlas (Hahmann *et al.*, 2020) provide wind speed estimates over large areas. However, they are built on large scale weather models and contain significant bias and uncertainty (Jourdier, 2020). Correcting the bias in these models can help wind farm developers and investors to make better decisions at the early stages of investment.

It is possible to infer the average wind speed from power generation data at nearby wind farms (Weiter *et al.*, 2019). Power generation data from existing wind farms is publicly available in some countries (ESIOS, 2022) and requires knowledge of the characteristics of the wind turbine, including the height of the turbine hub (see Figure 1). Sometimes this information is available on company websites, planning documents, or wind industry data providers (The Wind Power, 2023), but there are many missing wind farms and it is difficult to curate an accurate dataset.

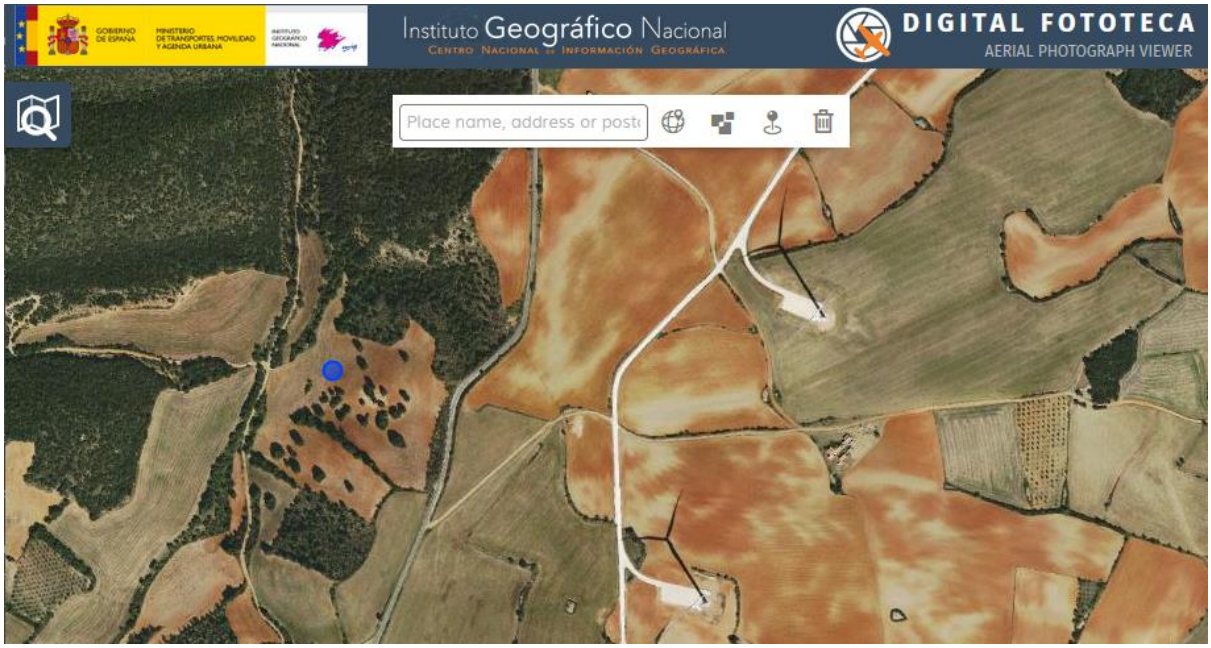

Figure 5: Two wind turbines shadows from the Sierra de Luna windfarm in Spain. The blue dot marks the centre of the nearest aerial photograph (Geográfica, 2022).

3. Data

3.1 Aerial photography

Spain has been chosen as the location for this project because of the quality of publicly available aerial imagery. The Instituto Geográfico Nacional was founded in 1870 to determine the shape and dimensions of the Earth, and today it maintains many datasets representing the real world (Instituto Geográfico Nacional, 2023). This includes aerial photography of Spain at a resolution of 0.25m or 0.15m per pixel, refreshed on a three year rolling programme. There is also a database of image metadata including timestamps (Geográfica, 2022).

Aerial imagery is distorted by the camera lens and there is extensive overlap between images. This distortion has a significant impact on the accuracy of measured distances. These issues are resolved in a set of orthophotos that combine the data from many aerial photographs and correct for the distortion. The orthophotos represent the correct position in a horizontal plan and allow for measurement of distances. Unfortunately there is no formal link between the orthophotos and the aerial photographs that have been used to create them, so this project will assume that each part of the orthophoto is derived from the nearest aerial photograph from recent years.

Aerial Imagery is available for all of Spain, however timestamp metadata is missing for 18% of our wind farm locations.

3.2 Digital elevation model

The Instituto Geográfico Nacional also provides digital elevation models for Spain (Geográfica, 2022). A digital terrain model with a 5m grid spacing is used to correct for any difference in elevation between the base of the turbine and the hub shadow.

3.3 Wind farm metadata

My employer, Aurora Energy Research, has supplied a curated list of 91 wind farm locations and hub heights in Spain, 76 of which can be matched to an image timestamp. These are based on data produced by The Wind Power (The Wind Power, 2023), and the hub heights have been checked by searching for corroborating evidence on the internet.

There are 61 wind farms in the initial dataset, with 36 allocated to the training set, 12 to the validation set, and 13 to the test set. 15 wind farms are reserved for use in experiments to measure the impact of active learning and adding more data. This allocation will persist through each stage of the project so that the test set can provide an unbiased measurement of the effectiveness of the approach.

4. Methodology

4.1 Turbine shadow model

The first model locates turbine shadows within a known wind farm. Measuring the shadow length requires high resolution imagery, but it would a very difficult and computationally expensive task to do this in such a large image. It would also increase the number of objects detected as false positives. Instead, an initial model was trained to detect the characteristic shape of a wind turbine shadow using scaled down images.

A data engineering pipeline was written to extract wind farm images from the orthophotos, each one being 2km x 2km centred on the coordinates provided. They were uploaded to the Roboflow platform and all the turbine shadows on the images were annotated with labels.

The default image size of 640 x 640px in YOLOv7 was used for this analysis. This was selected to meet computational constraints since it allows images to be processed in batches of 4 on a laptop with 4GB of RAM. To increase the performance of the model, Roboflow was used to split the image into 4 x 1km square tiles, before downscaling the image to 640 x 640px. This resulted in images with a resolution of 1.6m per pixel.

The baseline dataset has a maximum of 4 images per wind farm, with a few test images being dropped by the Roboflow platform. It contains 144 (60%) training images, 48 (20%) validation images and 49 (20%) test images. Performance was evaluated using the mAP:0.5-0.95 metric which is the best measure of the accuracy of the turbine position.

Figure 6: Predicted turbine shadows for the Plana de Jarreta wind farm. Note that the turbine in a field with a diagonal pattern was not detected.

An initial YOLOv7 model was trained using transfer learning with the existing model weights and 200 epochs of training. This number of epochs was selected because it showed that the model had some skill but was still under-fitting. This provided a good starting set of model weights, where the improvement from each experiment could be measured. A baseline was then created using an additional 200 epochs. Both these models use the default YOLOv7 hyperparameter configuration which already includes several data-centric techniques – colour transformation, translate, scale, flip and mosaic. These were retained in the default models because they make a significant contribution to model performance and training time.

Experiments were then run to measure performance with additional data-centric techniques. These were complimented with experiments to measure the impact of transformations included in YOLOv7 by default, by removing one of the default techniques at a time. Active learning was implemented by running a new set of images against the baseline model and selecting those where the objects were not correctly labelled. In a separate experiment a set of additional labels were also added.

Overfitting occurred in some of the experiments when run over 200 epochs. The was mitigated by using the best set of model weights from each run, as selected using the validation set. The selected model for each experiment was then evaluated using the test set of images.

4.2 Hub shadow model

The hub shadow can be measured by locating the centre of the base of the turbine and the hub shadow in each image, and calculating the distance between them.

A data engineering pipeline was written to extract hub shadow images based on the existing or predicted bounding boxes from the previous task. This produced images at full resolution that are a minimum of 640 x 640px, centred on the location of the shadow. If the labelled area was larger than 640px, then a square image was created at full resolution with a 10px border. This was downscaled to 640px later when generated by Roboflow.

The initial baseline dataset started with 1 image per wind farm, providing 36 (59%) training images, 12 (20%) validation images and 13 (21%) test images. Since this is such a small dataset, 2 additional images per wind farm were added to the test set to increase the consistency of evaluation, giving a total of 38 test images.

The remaining experimental methodology was the same as for the turbine shadow model, starting with an initial model with 600 epochs.

4.3 Estimate hub height

The hub height can be estimated from the length of the hub shadow and the position of the sun at the time of the photograph. This section describes the methodology including quality control measures and a correction for changes in ground level.

The hub shadow model estimates the position of the centre of the base and the hub shadow bounding boxes within the cropped image. The pixel size (either 0.25m or 0.15m) can be combined with the predicted labels to calculate the shadow length.

The aerial photography metadata contains the timestamp of most photographs, and is in Coordinated Universal Time (UTC). Initially this timestamp was retrieved once for each wind farm, but the accuracy of predictions was increased by finding the nearest timestamp for each turbine. This required the latitude and longitude of every image to be persisted through the project pipeline. A geospatial data frame of aerial photos was used to find the nearest image, rejecting turbines where data is missing because the nearest image is more than 3.1km away.

The aerial photograph timestamp was passed to the Skyfield astronomical software library (Rhodes, 2019) to estimate the relative positions of the sun and the earth. This was used to calculate the angle between the sun and the horizon, known as the altitude. It was also used to calculate the angle between the sun and grid north, known as the azimuth.

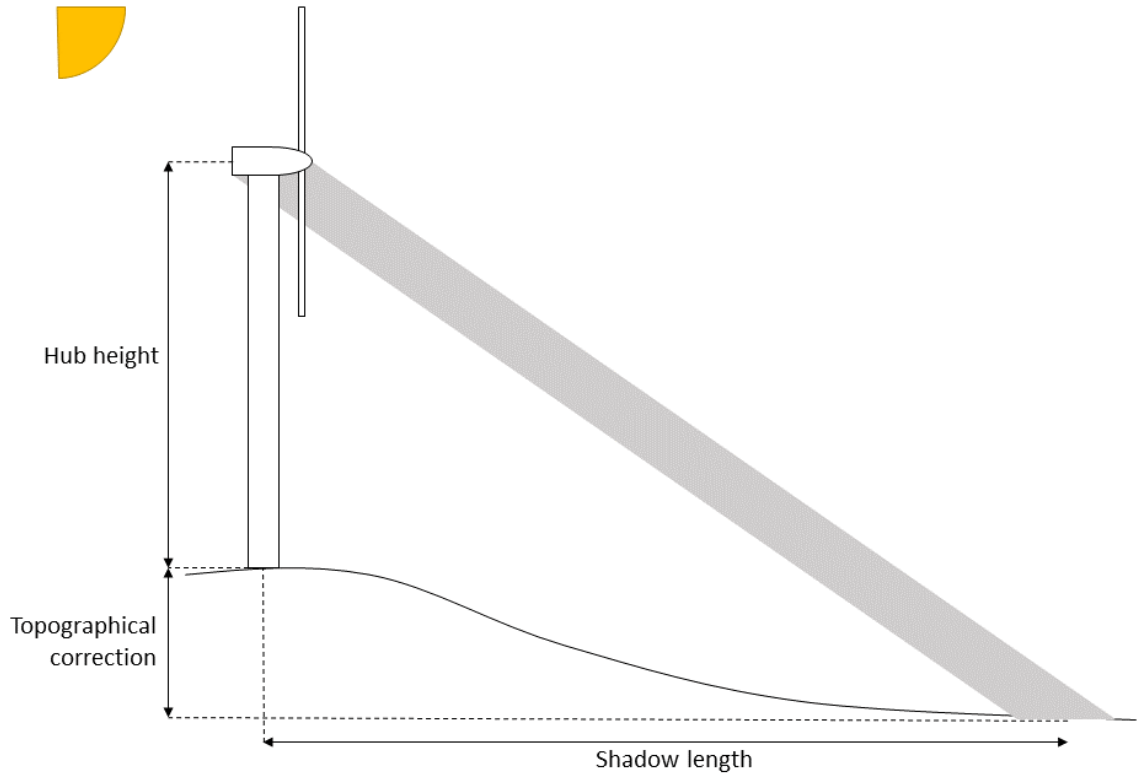

Figure 7: Topographical correction for estimated hub height

There were three quality control checks. If the models were unable to predict the location of the base or the hub shadow in an image then it cannot be used. If the models predict more than one of the same object in an image then that is also rejected. This can occur when turbines are closely spaced and the sun is low in the sky, but only affected a small number of turbines. The azimuth of the sun is also used to reject images where the timestamp does not correspond with the shadows in the image. The relative position of the base and hub shadow bounding boxes were used to estimate the azimuth of the sun, and if this differed from the timestamp azimuth by more than 5° the image was rejected.

The shadow height was calculated using simple trigonometry:

$$shadow\ height = shadow\ length * \tan(altitude)$$

The topographic correction shown in figure 7 can account for the difference in elevation between the base and the location of the hub shadow. This is common in this dataset because wind turbines are built on top of hills whenever possible to increase the energy that can be captured. An accurate latitude and longitude is calculated for each predicted base and shadow length. This is used to retrieve the elevation at each point from the manually downloaded digital elevation files. This is subtracted from the shadow height to get the estimated hub height.

The data was aggregated at the wind farm level before being analysed. This was done to avoid bias from wind farms with many turbines. Since each wind farm has a consistent sun

azimuth, hub shape, and terrain type, aggregating at the wind farm level gives a more representative set of data to be evaluated.

The hypothesis in RQ3 was tested on the set of known hub heights for each wind farm in the test dataset using a two-tailed t-test with a 5% significance level. To confirm whether a t-test is valid the data is plotted to look for outliers and a Shapiro-wilk test was carried out to check for normality.

5. Results

5.1 Turbine shadow task

The initial model was trained for 200 epochs and was used as a starting point for the other experiments. Each experiment ran for a further 200 epochs. The key experiment results are shown in Table 1, with additional results being added in Table 4 in Appendix 1.

The best experiment (T22) detects 93% of the turbines in the test images. It shows that data-centric techniques can improve our evaluation metric (mAP_0.5:0.95) from 33% to 45% for this task. This model adds additional images (T20) and a MixUp transformation (T16) to the defaults included in the baseline model.

Several data-centric transformations were tested, some of which are enabled by default. Removal of all default transformations (T3) reduced performance to 24%. This can mainly be attributed to the scale and flip transformations whose removal significantly reduces performance. Other defaults transformations actually increase performance when removed.

Of the six additional data transformations that were tested, only MixUp (T16) improves performance. Applying rotation or shear to the whole image or within the bounding box does not improve model performance.

Active learning (T20) with a small number of additional images that were not detected by the baseline model did not improve performance. Increasing the size of the dataset (T21) by 33% did improve performance by 2% to 43%.

The long training times for this dataset and configuration meant that experiment durations did not vary greatly. Even though two additional hours were required to label additional labels (T21), this duration is equivalent to training two data transformation experiments.

Table 1: Key experiment results for the turbine shadow task

Id	Experiment	Difference to baseline	mAP_0.5:0.95	Duration (hours)
T1	Initial model	Starting point for other experiments	0.33	-
T2	**Baseline**	None. Already includes default pre-processing for colour, translate, scale, flip and mosaic	**0.41**	4
T3	No transformations	Remove all default transformations (T4 – T8)	0.24	2.5
T4	No colour transformations	Remove Hue, Saturation, or Value (HSV) modifications	0.38	2.5
T5	No translate	0% (default is ±20%)	0.44	2.5
T7	No scale	0% (default is ±50%)	0.35	2.5
T8	No flip	0% (default is 50% left/right flip)	0.39	2.5
T9	No mosaic	0% (default is 100%)	0.39	2.5
T10	Rotation 10°	±10°	0.39	2.5
T12	Bounding box rotation 15°	Three copies of data with ±15° rotation generated by Roboflow	0.33	6.6
T13	Shear 10°	±10°	0.38	2.5
T15	Bounding box shear 10°	Three copies of data with ±10° shear generated by Roboflow	0.39	6.6
T16	MixUp	50% probability	0.43	2.5
T17	Noise	Three copies of data with 5% noise generated by Roboflow	0.38	3
T18	Histogram contrast equalization	Pre-processing of images to distribute differences in contrast	0.40	2.5
T20	Active learning	Add 11 (8%) additional training images that were not detected by the baseline model	0.41	4.5
T21	Larger dataset	Add 42 (22%) additional training and validating images	0.43	5.0
T22	**Best data-centric model**	Use additional labels, add MixUp, and retain default transformations.	**0.45**	2.5

5.2 Hub shadow task

The initial model was trained for 600 epochs and was used as a starting point for the other experiments. Each experiment ran for a further 200 epochs. The key experiment results are shown in Table 2, with additional results being shown in Table 5 in Appendix 2.

The best experiment (H26) uses active learning and the default data-centric transformations, detecting 74% of the images in the test set. It improves our evaluation metric (mAP_0.5:0.95) from 30% to 47%, which is a combination of 35% for the base and 59% for the hub shadow. Active learning identified 31 additional images where the base and/or hub shadow were not detected by the baseline model.

Several additional data-centric transformations improved results in isolation, but not when combined with the active learning dataset. Examples of this include adding 5% noise (H18 &

Table 2: Key experiment results for the hub shadow task

Id	Experiment	Difference to baseline	mAP_0.5:0.95			Duration (hours)
			All	base	hub	
H1	Initial model	Starting point for other experiments	0.24	0.18	0.30	-
H2	**Baseline**	None. Already includes default pre-processing for colour, translate, scale, flip and mosaic	**0.30**	**0.22**	**0.38**	**0.7**
H3	No transformations	Remove all default transformations (H4 – H8)	0.29	0.22	0.37	0.7
H4	No colour transformations	Remove Hue, Saturation, or Value (HSV) modifications	0.33	0.25	0.41	0.7
H5	No translate	0% (default is ±20%)	0.27	0.20	0.35	0.7
H6	No scale	0% (default is ±50%)	0.29	0.22	0.37	0.7
H7	No flip	0% (default is 50% left/right flip)	0.32	0.23	0.40	0.7
H8	No mosaic	0% (default is 100%)	0.31	0.19	0.44	0.7
H9	Rotation	±20°	0.19	0.11	0.28	0.7
H10	Bounding box 15° rotation	Three copies of data with ±15° rotation generated by Roboflow	0.37	0.22	0.52	1.8
H11	Shear	±20°	0.21	0.12	0.30	0.7
H14	Bounding box 15° vertical shear		0.39	0.27	0.51	1.8
H17	MixUp	50% probability	0.30	0.23	0.37	0.7
H19	10% Noise		0.39	0.26	0.53	1.8
H21	Cutout	Three copies of data with 10 squares of data masked out by Roboflow	0.36	0.24	0.48	2
H23	Paste in 50%	50% probability of other images being pasted over the sections of the image	0.30	0.18	0.68	0.7
H25	Adaptive contrast equalization	Pre-processing of images to locally distribute differences in contrast	0.30	0.23	0.37	1.8
H26	**Active learning**	Add 31 (86%) additional training images that were not detected by the baseline model	**0.47**	**0.35**	**0.59**	**3.5**
H27	Additional labels	Add 47 (98%) additional labels to the training and validation set	0.43	0.28	0.59	3
H28	All labels	Images from the active learning and additional dataset	0.33	0.25	0.40	2
H30	Active learning and 10% noise	Three copies of baseline and active learning images with 10% noise	0.29	0.26	0.33	2.8
H31	Active learning no colour modifications		0.35	0.25	0.45	2.8
H32	Active learning and 15% vertical shear		0.38	0.29	0.47	3.1
H33	Active learning and Cutout		0.27	0.17	0.36	3.1

H29), and removal of default colour modifications (H4 & H31). Both of these transformations showed strong improvements on the base object task in isolation, but not when used with the active learning dataset.

Removing all default transformations did not make a significant difference (H3). Translate and scale contributed to better performance; whilst flip, mosaic and colour transformation reduced it.

Rotation and shear of the full image significantly reduced model performance. Applying these transformations within the bounding boxes consistently improved performance on the main task, but not when combined with active learning (H32).

Adding additional images (H27) outperformed the baseline but not the active learning dataset. When combined with the active learning dataset (H28) this actually reduced performance.

Experiment durations were shorter than the turbine hub model because of the smaller dataset. This means that the additional time to prepare with additional labels and train (H27) is equivalent to running five other data transformation experiments.

5.3 Hub height estimation

Using the best hub shadow model, a base and a hub shadow object were both identified in 74% of the 151 turbine images in the test set. 13% of the images were rejected by quality checks, primarily when the tower shadow did not align with the calculated angle of the sun at the time. Rodera Alta was affected by this tower shadow alignment issue where investigation confirmed that the wrong timestamps were being used. In this case there was an overlap between different flights, with some aerial photographs being taken closer to the to the wind farm that were not used in the creation of the orthophoto.

Eight of the eleven test wind farms have hub height errors within 5m. The remaining three, Hoya Gonzalo, Serra da Loba and Xiabre have hub height errors between 5 and 10m. A one-sample two-tailed t-test gives a p-value of 0.004 for the RQ3 hypothesis that “the average error of the estimated hub height is less than 5m”. Inspection of the histogram and a Shapiro-Wilk test ($p = 0.15$) confirm that the test data is from a normal distribution. This is small sample size of eleven wind farms, so the test does not have much statistical power.

The training data gives different results because the presence of outliers is more pronounced, as shown in Figure 8 and Appendix A.3. This data is not normally distributed, with a p-value for the Shapiro-Wilk test that is approximately zero. The t-test gives a p-value of 0.093, so we fail to reject the null hypothesis when applied to the training set.

These outliers suggest that there are systematic errors that could be addressed and that it is not be appropriate to assume a normal distribution of errors. A simpler evaluation is that 27% of the test wind farms have hub height errors greater than 5m. In the training set, this increases to 14 out of 49 training wind farms (29%).

Table 3: Estimated hub heights for the test wind farms

Wind farm	Number of turbines	Valid estimates	Actual hub height	Estimated hub height	Hub height error
Ausine	2	100%	80	81.1	1.1
Becerril	4	100%	83.8	87.2	3.5
Hoya Gonzalo	13	54%	50	55.4	5.4
Malpica	6	33%	78.3	81.3	3
Monte Seixo	20	60%	55	55.1	0.1
Palomarejo	5	20%	80	78.5	-1.5
Paxareiras Montevós	28	71%	35	34.8	-0.2
Plana de Jarreta	39	82%	55	55.8	0.8
Requeixo	13	69%	60	60.9	0.9
Rodera Alta	4	0%	78		
Serra da Loba	11	45%	78	68.2	-9.8
Xiabre	6	33%	87	80.8	-6.2
	151	**64%**			

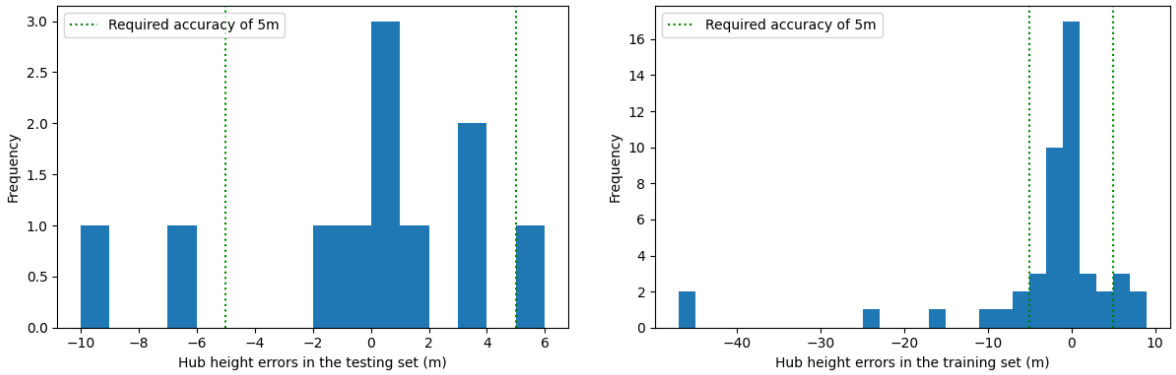

Figure 8: Hub height errors in the test and training data sets

6. Discussion

6.1 Data-centric transformations

Both computer vision models show the importance of data-centric transformations in model performance. The default transformations in YOLO_v7 contribute strongly to the turbine shadow model and have a small impact on the hub shadow model. All default transformations were retained in both of the final models. These default transformations were probably chosen to optimize for ground-level photography and videos that were shot where the camera pointed parallel to the ground, rather than perpendicular as in aerial imagery. These results show that the default data-centric configuration is well suited to aerial photography.

Different data-centric techniques are optimal for different datasets. The turbine shadow model benefitted from additional labels and the MixUp transformation. The hub shadow model benefitted most from the new labels selected by active learning. For this reason data

scientists should always plan time to tune the data hyperparameters for their computer vision models.

Embedding data transformations to the training set in the YOLO_v7 codebase is a significant advantage, even without the default settings. It avoids the availability bias against data-centric AI by making it quick and easy to tune these values, even allowing the data hyperparameter tuning to be automated. In software engineering it is well established that testing is easier and more effective when closely coupled with the code that is being tested, this is why unit testing is such important. In a similar way, embedding data-centric transformations in the model code make it much easier to identify the best transformations for a new dataset. This advantage is likely to be adopted by other computer vision model, since it will be harder for them to compete without it.

The two computer vision models perform well for the task of estimating hub height. The turbine model achieved 45% mAP_0.5:0.95 and the hub shadow model achieved an average of 47%. This can be compared to YOLOv7's 57% for all objects (Wang, Bochkovskiy and Liao, 2022), which was achieved on a far larger dataset. This confirms that good results can be achieved in computer vision with small datasets, although there is still room for improvement.

The purpose of the turbine shadow model is to detect all the turbines. It achieved an mAP_0.5 of 88%. Many turbines were labelled, however these were often quite similar since they belonged to the same wind farm and would have the same shadow angle, profile, and background terrain. More diversity in the number of wind farms represented would help here.

The hub shadow model performed better when detecting the hub shadow object (59%) than the turbine base object (35%). The base is a difficult object to detect because it is circular and half of it is systematically occluded by the tower, with the same segment of the base often being occluded for adjacent turbines. This understanding of the data can explain why active learning was so influential on this dataset. Providing additional labels with different parts of the base occluded is needed to help the model achieve better performance.

These are small datasets so it not surprising that data-centric techniques are important. Without the ability to average weights over many images, the quality of individual images and the ability generalize a model without overfitting become crucial. Results for both tasks demonstrate that they would benefit from further gathering and labelling of images. The stronger performance of active learning in the hub-shadow results show that this is a technique that should always be considered when data needs to be manually labelled. This shows that an ML Ops platform should always be used when labelling new datasets to make it easier to gather and consistently label new images. Adopting a data-centric mindset will

Figure 9: Example of alignment between the wind and the shadow at Lezuza

help the Data Scientist needs to focus sufficient attention on the activities that will improve performance.

6.2 Hub height estimation

There is conflicting evidence that the average hub height error is less than 5m. The t-test on the test data set proves that the population mean of the test set is less than 5m with a p-value of 0.004. However, the sample size is small and has low statistical power. Also three of the eleven wind farms in the test set are outside the 5m threshold.

The training set contains 49 wind farms where the hub height is known, and even though it should be more accurate because this data was used to train the hub shadow model, the t-test fails to reject the null hypothesis with a value of 0.093. This is due to 14 estimates having an error greater than 5m, with 5 of these having an error greater than 10m.

To achieve these results, three additional features needed to be implemented that were not originally envisaged in the scope of the project. The first is that the photo timestamp was retrieved for each individual turbine rather than a single value for each wind farm. Secondly, turbines are rejected where the tower azimuth does not match the azimuth of the sun at

the given time. Finally, a topology correction is applied using digital elevation models to account for the difference between the elevation of the base and the hub shadow.

The outliers in results indicate one or more remaining sources of error in the method. One cause is errors in the ground truth for the hub height. Several of these errors have already been found and corrected, and more may remain, especially in the training set. Hub height information is difficult to find and needs to be matched to a set of coordinates. It also assumes that the turbines were actually built to the heights that were approved during the planning process. An example is the Ourol wind farm which was originally in the test set, but further investigation showed that the hub height was actually for a newer wind farm built 3km to the East. Such outliers can be safely removed from the dataset if the information is shown to be in error.

A second cause of error could occur when the wind aligns with the shadow, as shown in figure 9. The centre of the tower is usually positioned in front of the centre of the hub, in order to balance the weight of the blades. When the wind happens to align with the azimuth of the sun then this mismatch can cause the shadow length to be over or underestimated. An additional computer vision model would be needed to measure the wind direction. This could be done using image segmentation to detect the blades which would require a more complicated form of labelling. This error is unlikely to be more than 5m in magnitude, so could not account for the larger outliers in the training data set.

7. Conclusion

Even though they use the same images, the two use cases needed different data-centric techniques to achieve the best performance. The turbine shadow model benefitted by adding additional data and by using MixUp to help the model generalize to turbines in any part of the image. The hub shadow model needed active learning to identify different angles of occlusion between the tower and the base. These specific data needs can only be identified by a data-centric approach.

A data-centric approach is important when building a predictive model on small datasets. Effective data-centric tooling is needed, not just in ML Ops platforms. Embedding the transformation of training images into model code is an efficient way to improve performance, and is likely to be required for any successful computer vision framework. It enables data hyperparameter tuning, making it intuitive, effective and automatable. Data-centric improvements are already being implemented across models and platforms and this trend is likely to build in future.

Andrew Ng is right to widely promote the importance of a data-centric mindset across the data science community. Most data science projects have less labelled data than is optimal, and so teams with the tools and discipline to optimize the data that they have will achieve the best results. To do this data scientists need to be aware of the natural bias for model-

centric techniques, so they can allow for the systematic adoption of data-centric techniques in project plans.

Although this project does not prove that the errors in estimated hub height are within 5m, estimates at 70% of wind farms did fall within this threshold. Achieving these results required additional functionality to accurately identify the correct aerial photo for each wind turbine, and to correct for differences in elevation between the base and the hub shadow. Several of the initial outliers found by this project were investigated and found to be errors in the ground truth of hub heights previously retrieved from the internet. This allowed this data to be corrected for other projects and shows the importance of an independent method to measure wind turbine hub height.

Word count: 7247 words

References

[1] Bochkovskiy, A., Wang, C.-Y. and Liao, H.-Y.M. (2020) 'YOLOv4: Optimal Speed and Accuracy of Object Detection'. arXiv. Available at: https://doi.org/10.48550/arXiv.2004.10934.

[2] *COCO - Common Objects in Context* (2023). Available at: https://cocodataset.org/#detection-eval (Accessed: 5 March 2023).

[3] *Data-centric AI Resource Hub* (2022) *Data-centric AI Resource Hub*. Available at: https://datacentricai.org/ (Accessed: 29 January 2023).

[4] DeVries, T. and Taylor, G.W. (2017) 'Improved Regularization of Convolutional Neural Networks with Cutout'. arXiv. Available at: https://doi.org/10.48550/arXiv.1708.04552.

[5] ESIOS (2022) *About ESIOS electricity*. Available at: https://www.esios.ree.es/en/about-esios (Accessed: 29 January 2023).

[6] Everingham, M. *et al.* (2010) 'The Pascal Visual Object Classes (VOC) Challenge', *International Journal of Computer Vision*, 88(2), pp. 303–338. Available at: https://doi.org/10.1007/s11263-009-0275-4.

[7] Figueroa, R.L. *et al.* (2012) 'Predicting sample size required for classification performance', *BMC Medical Informatics and Decision Making*, 12(1), p. 8. Available at: https://doi.org/10.1186/1472-6947-12-8.

[8] Geográfica, O.A.C.N. de I. (2022) *Centro de Descargas del CNIG (IGN)*, *Centro de Descargas del CNIG*. Available at: http://centrodedescargas.cnig.es (Accessed: 13 November 2022).

[9] Ghiasi, G., Lin, T.-Y. and Le, Q.V. (2018) 'DropBlock: A regularization method for convolutional networks'. arXiv. Available at: https://doi.org/10.48550/arXiv.1810.12890.

[10] Hahmann, A.N. *et al.* (2020) 'The making of the New European Wind Atlas – Part 1: Model sensitivity', *Geoscientific Model Development*, 13(10), pp. 5053–5078. Available at: https://doi.org/10.5194/gmd-13-5053-2020.

[11] Hamid, O.H. (2022) 'From Model-Centric to Data-Centric AI: A Paradigm Shift or Rather a Complementary Approach?', in *2022 8th International Conference on Information Technology Trends (ITT)*. *2022 8th International Conference on Information Technology Trends (ITT)*, pp. 196–199. Available at: https://doi.org/10.1109/ITT56123.2022.9863935.

[12] Jourdier, B. (2020) 'Evaluation of ERA5, MERRA-2, COSMO-REA6, NEWA and AROME to simulate wind power production over France', in *Advances in Science and Research*. *19th EMS Annual Meeting: European Conference for Applied Meteorology and Climatology 2019* -, Copernicus GmbH, pp. 63–77. Available at: https://doi.org/10.5194/asr-17-63-2020.

[13] Keshari, R. *et al.* (2018) 'Learning Structure and Strength of CNN Filters for Small Sample Size Training', in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9349–9358. Available at: https://doi.org/10.1109/CVPR.2018.00974.

[14] Lewis, D.D. and Gale, W.A. (1994) 'A Sequential Algorithm for Training Text Classifiers', in B.W. Croft and C.J. van Rijsbergen (eds) *SIGIR '94*. London: Springer, pp. 3–12. Available at: https://doi.org/10.1007/978-1-4471-2099-5_1.

[15] Li, P. *et al.* (2021) 'CleanML: A Study for Evaluating the Impact of Data Cleaning on ML Classification Tasks'. arXiv. Available at: https://doi.org/10.48550/arXiv.1904.09483.

[16] Instituto Geográfico Nacional (2023) *Geoportal oficial del Instituto Geográfico Nacional de España*. Available at: http://www.ign.es (Accessed: 19 February 2023).

[17] Ng, A. (2021) 'A chat with andrew on mlops: From model-centric to data-centric ai'. DeepLearningAI. [Online]. Available: https://www.youtube.com/watch?v=06-AZXmwHjot=1607s (Accessed: 13 November 2022).

[18] Pizer, S.M. *et al.* (1987) 'Adaptive histogram equalization and its variations', *Computer Vision, Graphics, and Image Processing*, 39(3), pp. 355–368. Available at: https://doi.org/10.1016/S0734-189X(87)80186-X.

[19] Redmon, J. *et al.* (2016) 'You Only Look Once: Unified, Real-Time Object Detection', in. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788. Available at: https://www.cv-foundation.org/openaccess/content_cvpr_2016/html/Redmon_You_Only_Look_CVPR_2016_paper.html (Accessed: 30 October 2022).

[20] Ren, S. *et al.* (2015) 'Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks', in *Advances in Neural Information Processing Systems*. Curran Associates, Inc. Available at: https://proceedings.neurips.cc/paper/2015/hash/14bfa6bb14875e45bba028a21ed38046-Abstract.html (Accessed: 19 November 2022).

[21] Rhodes, B. (2019) 'Skyfield: High precision research-grade positions for planets and Earth satellites generator', *Astrophysics Source Code Library*, p. ascl:1907.024.

[22] Rosebrock, A. (2016) 'Measuring distance between objects in an image with OpenCV', *PyImageSearch*, 4 April. Available at: https://pyimagesearch.com/2016/04/04/measuring-distance-between-objects-in-an-image-with-opencv/ (Accessed: 26 November 2022).

[23] Sambasivan, N. *et al.* (2021) '"Everyone wants to do the model work, not the data work": Data Cascades in High-Stakes AI', in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*.

[24] Schneider, M. *et al.* (2022) 'A wind atlas for Germany and the effect of remodeling', *Meteorologische Zeitschrift*, pp. 117–130. Available at: https://doi.org/10.1127/metz/2022/1102.

[25] Settles, B. (2009) *Active Learning Literature Survey*. Available at: https://www.semanticscholar.org/paper/Active-Learning-Literature-Survey-Settles/818826f356444f3daa3447755bf63f171f39ec47 (Accessed: 5 March 2023).

[26] Sun, D. *et al.* (2020) 'A Scale Balanced Loss for Bounding Box Regression', *IEEE Access*, 8, pp. 108438–108448. Available at: https://doi.org/10.1109/ACCESS.2020.3001234.

[27] Synced (2020) 'YOLO Creator Joseph Redmon Stopped CV Research Due to Ethical Concerns', *SyncedReview*, 24 February. Available at: https://medium.com/syncedreview/yolo-creator-says-he-stopped-cv-research-due-to-ethical-concerns-b55a291ebb29 (Accessed: 12 February 2023).

[28] *The Wind Power* (2023). Available at: https://www.thewindpower.net/ (Accessed: 19 February 2023).

[29] Tordoff, S. (2013) *How to plan the perfect wind measurement campaign*. Available at: https://www.windpowermonthly.com/article/1172038/plan-perfect-wind-measurement-campaign?utm_source=website&utm_medium=social (Accessed: 19 February 2023).

[30] Wagstaff, K. (2012) 'Machine Learning that Matters'. arXiv. Available at: https://doi.org/10.48550/arXiv.1206.4656.

[31] Wang, C.-Y., Bochkovskiy, A. and Liao, H.-Y.M. (2022) 'YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors'. arXiv. Available at: https://doi.org/10.48550/arXiv.2207.02696.

[32] Wang, Y. *et al.* (2022) 'Remote sensing image super-resolution and object detection: Benchmark and state of the art', *Expert Systems with Applications*, 197, p. 116793. Available at: https://doi.org/10.1016/j.eswa.2022.116793.

[33] Weiter, A. *et al.* (2019) 'Electricity production by wind turbines as a means for the verification of wind simulations', *Meteorologische Zeitschrift*, pp. 69–77. Available at: https://doi.org/10.1127/metz/2019/0924.

[34] Zoph, B. *et al.* (2019) 'Learning Data Augmentation Strategies for Object Detection'. arXiv. Available at: https://doi.org/10.48550/arXiv.1906.11172.

Appendices

Appendix 1: Additional turbine shadow results

This appendix contains results from all the experiment that were carried out.

Table 4: All experiment results for the turbine shadow task

Id	Experiment	Difference to baseline	mAP_ 0.5:0.95	Duration (hours)
T1	Initial model	Starting point for other experiments	0.33	-
T2	**Baseline**	None. Already includes default pre-processing for colour, translate, scale, flip and mosaic	**0.41**	4
T3	No transformations	Remove all default transformations (T4 – T8)	0.24	2.5
T4	No colour transformations	Remove Hue, Saturation, or Value (HSV) modifications	0.38	2.5
T5	No translate	0% (default is ±20%)	0.44	2.5
T6	Translate	5%	0.40	2.5
T7	No scale	0% (default is ±50%)	0.35	2.5
T8	No flip	0% (default is 50% left/right flip)	0.39	2.5
T9	No mosaic	0% (default is 100%)	0.39	2.5
T10	Rotation 10°	±10°	0.39	2.5
T11	Rotation 20°	±20°	0.29	2.5
T12	Bounding box rotation 15°	Three copies of data with ±15° rotation generated by Roboflow	0.33	6.6
T13	Shear 10°	±10°	0.38	2.5
T14	Shear 20°	±20°	0.28	2.5
T15	Bounding box shear 10°	Three copies of data with ±10° shear generated by Roboflow	0.39	6.6
T16	MixUp	50% probability	0.43	2.5
T17	Noise	Three copies of data with 5% noise generated by Roboflow	0.38	3
T18	Histogram contrast equalization	Pre-processing of images to distribute differences in contrast	0.40	2.5
T19	Adaptive contrast equalization	Pre-processing of images to locally distribute differences in contrast	0.39	2.5
T20	Active learning	Add 11 (8%) additional training images that were not detected by the baseline model	0.41	4.5
T21	Larger dataset	Add 42 (22%) additional training and validating images	0.43	5.0
T22	**Best data-centric model**	Use additional labels, add MixUp, and retain default transformations.	**0.45**	2.5

Appendix 2: Additional hub shadow results

Table 5: All experiment results for the hub shadow task

Id	Experiment	Difference to baseline	mAP_0.5:0.95			Duration (hours)
			All	base	hub	
H1	Initial model	Starting point for other experiments	0.24	0.18	0.30	-
H2	**Baseline**	None. Already includes default pre-processing for colour, translate, scale, flip and mosaic	**0.30**	**0.22**	**0.38**	**0.7**
H3	No transformations	Remove all default transformations (H4 – H8)	0.29	0.22	0.37	0.7
H4	No colour transformations	Remove Hue, Saturation, or Value (HSV) modifications	0.33	0.25	0.41	0.7
H5	No translate	0% (default is ±20%)	0.27	0.20	0.35	0.7
H6	No scale	0% (default is ±50%)	0.29	0.22	0.37	0.7
H7	No flip	0% (default is 50% left/right flip)	0.32	0.23	0.40	0.7
H8	No mosaic	0% (default is 100%)	0.31	0.19	0.44	0.7
H9	Rotation	±20°	0.19	0.11	0.28	0.7
H10	Bounding box 15° rotation	Three copies of data with ±15° rotation generated by Roboflow	0.37	0.22	0.52	1.8
H11	Shear	±20°	0.21	0.12	0.30	0.7
H12	Bounding box 15° horizontal shear	Three copies of data with ±15° shear generated by Roboflow	0.39	0.23	0.55	1.8
H13	Bounding box 10° vertical shear		0.37	0.24	0.50	1.8
H14	Bounding box 15° vertical shear		0.39	0.27	0.51	1.8
H15	Bounding box 20° vertical shear		0.39	0.26	0.53	1.8
H16	Bounding box 15° shear on both axes		0.35	0.26	0.44	1.8
H17	MixUp	50% probability	0.30	0.23	0.37	0.7
H18	5% Noise	Three copies of data generated by Roboflow with 5% noise	0.39	0.27	0.51	1.8
H19	10% Noise		0.39	0.26	0.53	1.8
H20	15% Noise		0.37	0.24	0.50	1.8
H21	Cutout	Three copies of data with 10 squares of data masked out by Roboflow	0.36	0.24	0.48	2
H22	Paste in 10%	10% probability of other images being pasted over the sections of the image	0.30	0.23	0.65	0.7
H23	Paste in 50%	50% probability of other images being pasted over the sections of the image	0.30	0.18	0.68	0.7
H24	Histogram contrast equalization	Pre-processing of images to distribute differences in contrast	0.29	0.18	0.39	1.8
H25	Adaptive contrast equalization	Pre-processing of images to locally distribute differences in contrast	0.30	0.23	0.37	1.8
H26	**Active learning**	Add 31 (86%) additional training images that were not detected by the baseline model	**0.47**	**0.35**	**0.59**	**3.5**

H27	Additional labels	Add 47 (98%) additional labels to the training and validation set	0.43	0.28	0.59	3
H28	All labels	Images from the active learning and additional dataset	0.33	0.25	0.40	2
H29	Active learning and 5% noise	Three copies of baseline and active learning images with 5% noise	0.40	0.32	0.48	3.2
H30	Active learning and 10% noise	Combining the best transformations	0.29	0.26	0.33	2.8
H31	Active learning no colour modifications		0.35	0.25	0.45	2.8
H32	Active learning and 15% vertical shear		0.38	0.29	0.47	3.1
H33	Active learning and Cutout		0.27	0.17	0.36	3.1
H34	Active learning with 100 more epochs	Additional epochs led to overfitting	0.22	0.17	0.27	1.4

Appendix 3: Estimated hub height for the training set

Table 6: Estimated hub heights for the training wind farms

Wind farm	Number of turbines	Valid estimates	Actual hub height	Estimated hub height	Hub height error
ablitas	3	67%	120	120.8	0.8
adrano	19	53%	50	44.2	-5.8
agualla_aguilar	15	73%	58	57.5	-0.5
alta_anoia	9	67%	80	78.6	-1.4
balson	5	20%	119	73.4	-45.6
bon_vent	3	100%	80	83.2	3.2
brulles	12	58%	105	104.5	-0.5
caacoloma	7	71%	85	82.3	-2.7
cabanillas	4	75%	120	120.4	0.4
cabezo_san	26	8%	48	55.4	7.4
carracha	32	75%	48	56.1	8.1
coll_moro	5	100%	100	96.2	-3.8
corbera	8	50%	80	80.7	0.7
dehesica	8	100%	80	80.3	0.3
fatarella	7	57%	80	82.7	2.7
lezuza	7	86%	75	81.2	6.3
magal	14	50%	63	54.6	-8.4
magaz	15	73%	80	70.2	-9.8
marmellar	8	75%	61	61	0
montouto	15	80%	49	55.4	6.4
muela_norte	24	88%	55	54.8	-0.2
munera	6	100%	80	78.5	-1.5
nava	4	100%	107	83.8	-23.2
navica	9	100%	80	79.2	-0.8
parc_ecovent	26	73%	60	59.9	-0.1
pujalt	7	71%	80	77.5	-2.5
rapos	18	56%	61	61.1	0.1

Wind farm	Number of turbines	Valid estimates	Actual hub height	Estimated hub height	Hub height error
redondal	12	42%	55	54.6	-0.4
rinconada	6	33%	85	83.4	-1.6
roden	2	50%	82	79.9	-2.1
romeral	12	33%	100	54.6	-45.4
santo_cristo	16	94%	80	79.9	-0.1
sarda	4	25%	85	83.3	-1.7
sardon	11	18%	55	55.4	0.3
sierra_luna	7	57%	84	82.6	-1.3
sierra_ministra	6	67%	80	79.1	-0.9
tablares	8	88%	119	102.3	-16.7
torre_madrina	3	100%	100	96.9	-3.1
valdeconejos	17	47%	55	49.3	-5.7
valdivia	11	27%	80	76.5	-3.5
valiente	7	100%	93	92.6	-0.4
veciana	5	40%	80	82.6	2.6
vilalba_dels	6	50%	100	98.6	-1.4
viudo	17	88%	78	81	3
ablitas	3	67%	120	120.8	0.8
adrano	19	53%	50	44.2	-5.8
agualla_aguilar	15	73%	58	57.5	-0.5
alta_anoia	9	67%	80	78.6	-1.4
	466	**64%**			