Vortex and Core Detection using Computer Vision and Machine Learning Methods

Zhenguo Xu, Ayush Maria, Kahina Chelli,
Thibaut Dumouchel De Premare, Xabadin Bilbao,
Christopher Petit, Robert Zoumboulis-Airey, Irene Moulitsas,
Tom Teschner, Seemal Asif and Jun Li*

*School of Aerospace, Transport and Manufacturing, Cranfield University,
Cranfield, Bedfordshire MK43 0AL, UK
E-mail: zhenguo.xu.545@cranfield.ac.uk; Ayush.Maria.256@cranfield.ac.uk;
Kahina.Chelli.248@cranfield.ac.uk; t.dumoucheldepremare.684@cranfield.ac.uk;
Xabadin.Bilbao.977@cranfield.ac.uk; c.d.petit.904@cranfield.ac.uk;
Robert.Zoumboulis-Airey@cranfield.ac.uk; i.moulitsas@cranfield.ac.uk;
Tom.Teschner@cranfield.ac.uk; s.asif@cranfield.ac.uk; Jun.Li@cranfield.ac.uk
Corresponding Author

Abstract

The identification of vortices and cores is crucial for understanding airflow motion in aerodynamics. Currently, numerous methods in Computer Vision and Machine Learning exist for detecting vortices and cores. This research develops a comprehensive framework by combining classic Computer Vision and state-of-the-art Machine Learning techniques for vortex and core detection. It enhances a CNN-based method using Computer Vision algorithms for Feature Engineering and then adopts an Ensemble Learning approach for vortex core classification, through which false positives, false negatives, and computational costs are reduced. Specifically, four features, i.e., Contour Area, Aspect Ratio, Area Difference, and Moment Centre, are employed

to identify vortex regions using YOLOv5s, followed by a hard voting classifier based on Random Forest, Adaptive Boosting, and Xtreme Gradient Boosting algorithms for vortex core detection. This novel approach differs from traditional Computer Vision approaches using mathematical variables and image features such as HAAR and SIFT for vortex core detection. The findings show that vortices are detected with a high degree of statistical confidence by a fine-tuned YOLOv5s model, and the integrated technique produces an accuracy score of 97.56% in detecting vortex cores conducted on a total of 133 images generated from a rotor blade NACA0012 simulation. Future work will focus on framework generalisation with a larger and more diverse dataset and intelligent threshold development for more efficient vortex and core detection.

Keywords: Computational fluid dynamics, rotor blade, mesh, ensemble learning, hard voting.

1 Introduction

Understanding the airflow around specific objects (e.g., cars, buildings, and turbo-machinery) known as aerodynamics is of great interest in many technical fields as the product performance strongly depends on it [11]. For instance, studying the airflow over a rotor blade is important because it improves the turbine's performance, efficiency and longevity etc. Depending on the environment (e.g., in a wind farm) the airflow over blades can be unstable and turbulent. Being able to understand how vortices behave and move through space by identifying their formation may allow designers to improve the turbine performance. The detection of vortices and vortex cores is also the foundation of numerous aerodynamic subjects, serving as a catalyst for in-depth research that builds upon the study of vortices.

However, the detection of vortices and their cores in airflow with instability and turbulence is not straightforward. Sujudi and Haimes [15] develop an algorithm to identify vortex cores in 3D discrete vector fields. They use the distributed environment of pV3 and linear interpolation to find the trajectories of vortex cores by employing cell-by-cell processing. Unfortunately, the research only considers the vortex core trajectory, ignoring the effect of vortex strength and mesh roughness on the accuracy of the results, resulting in inaccurate vortex core detection results. Biswas et al. [5] implements a voting algorithm based on Machine Learning classifiers to determine the proximity of points from vortex zones. This study is effective to identify

vortex regions but not to detect vortex cores. Azmi and Baliga [4] have shown that ensemble approaches are more effective and yield superior results compared to a single decision tree in classification tasks. Furthermore, it is found that ensemble approaches utilizing boosting methods such as XGBoost and AdaBoost outperform bagging methods such as Random Forest when there is little noise in the data [6, 14]. However, the application of ensemble approaches in vortex and core detection remains relatively scarce in practice.

This study aims to develop a comprehensive framework to detect the vortices and cores, and subsequently improve the detection accuracy and reduce the computational cost by integrating Computer Vision with Machine Learning methods. This framework uses a voting ensemble classifier to determine the proximity of points in vortex zones. The inputs of the framework are four vortex features obtained from Computer Vision methods and YOLOv5s.

The remainder of this report is organised as follows: Section 2 reviews the research on vortex and core detection; Section 3 illustrates the framework and its implementation steps; Section 4 presents the results and evaluates the proposed methods; Section 5 analyses and discusses the results, limitations, and recommendations; Section 6 concludes the work.

2 Literature Review

A vortex is a flow region in a fluid where it revolves around an axis. Vortices play a crucial role in aerodynamics product design as they often interact significantly with one another, leading to an increase in drag caused by friction with the fluid particles. It is therefore important to identify vortices and cores in aerodynamics.

There are solutions available such as using numerical information and Machine Learning methods to identify vortices and cores. For example, Levit and Lasinski [9] develop the TOPO software system that numerically analyses topological aspects of a 3D vector field for vortex core detection and visualisation. Sujudi and Haimes [15] develops an algorithm to identify vortex cores in 3D discrete vector fields based on critical point theory and implement it in the pV3 visualisation package. Jiang et al. [7] present a list of vortex core detection methods including Eigenvector, Swirl Parameter Method, and Lambda2 based on Computer Vision. However, the conventional Computer Vision and Machine learning approaches are limited in their accuracy, adaptability, and scalability due to the sizes and forms of flow fields.

Recently, Deep Learning methods such as Convolutional Neural Networks (CNN) are applied for vortex and core detection. Majchrzak et al. [11]

train a Keras-RetinaNet for vortex core detection, with a total of 1000 images captured by slicing in the flow direction (X-axis) in increments of 10 meters. The authors assert that the proposed method yields good results within a short time frame, although specific evaluation metrics are not provided. Furthermore, the method does not require additional data beyond the velocity field. Luo et al. [10] use YOLOv3, an object detection model based on CNN architecture, trained using the self-built vortex dataset from flow field photographs. Experiments show that the CNN-based technique outperforms traditional methods. Abolholl et al. [2] use YOLOv3 to detect flow structures directly from streamline plots. The results show that the approach reduces the number of false positives and negatives while removing the need for a cutoff. Abras and Hariharan [3] develop a CNN model to classify no-vortex or vortex in a large-scale 3D fluid-dynamic simulation. The model is trained using a collection of grey-scale vortex/no-vortex images with entropy contour plots acquired from a variety of rotor wake data sources. Its model results show that the performance is on par with humans in detecting vortices, but unable to confidently make predictions as the vortices start to break up. Xiong et al. propose a novel learning-based framework consisting of two Deep Learning Neural Networks, i.e., a vortex representation network to identify the Lagrangian vortices and a vortex interaction network to learn the underlying governing dynamics. The results show the method's efficacy in generating highly accurate predictions with low computational costs.

Deep learning algorithms can be time-consuming and computationally complex for vortex and core detection. To address these challenges, hybrid approaches are suggested. Wang et al. [16] introduces a unique vortex detection approach based on Convolutional Neural Networks-Extreme Learning Machine (CNN-ELM) to address the issue. This method converts the vortex detection from the flow field into a binary classification problem. Studies show that the method is effective and improves or supplements the existing methods. Ensemble Learning approaches for classification [4, 8, 13] combining the performance of multiple classification models based on metaheuristic optimisation techniques such as voting or boosting tend to reduce bias and provide more precise prediction than individual classification models. Biswas et al. [5] present a framework to investigate the uncertainty of four vortex detection methods and integrate them based on a voting algorithm to conduct vortex analysis and detection in a more robust manner. It first categorises the vortex zones and clusters them according to their geographical placements, and then identifies the remaining locations as either vortex or non-vortex

depending on their proximity from their nearest vortex cluster. The research tests the strategy on different datasets and time steps and finds that the method improves the model accuracy. Abolholl et al. [1] propose a hybrid approach based on Computer Vision and Machine Learning. Firstly, a CNN is trained to detect vortex locations in line integral convolution (LIC)-based streamline plots. Then, a Machine Learning layer is added to the CNN to locate the vortex cores. The experiments show that the training time is reduced by a factor of two with high prediction accuracy. However, as the specific details regarding the data processing pipeline and model architecture are not provided, the approach's adaptability and scalability for other vortex scenarios are questionable.

The literature review of various studies on vortex and core detection in fluid dynamics shows that most recent studies commonly use Deep Learning algorithms such as customised CNNs or YOLO models. These approaches are efficient in detecting vortex regions but not as efficient and accurate in detecting vortex core locations. Ensemble Learning approaches combining multiple classification models tend to provide more precise predictions than individual models. This study aims to develop a comprehensive framework with detailed data processing pipelines for vortex and core detection in fluid dynamics. The framework employs Machine Vision, Deep Learning based on YOLOv5s, and Ensemble Learning utilizing Random Forest, AdaBoost, and XGBoost classifiers. A 3D vortex core flow is also visualized.

3 Methodology

The framework based on CNN and Machine Learning methods for the detection of vortices and their cores is shown in Figure 1:

First, a CFD (Computational Fluid Dynamics) simulation generates raw data based on a defined case. The raw data is then visualised, preprocessed, and labelled for a YOLO model to detect vortices. Subsequently, the detected vortices and parameters are put through a Computer Vision based Feature Engineering pipeline to generate features which are then applied to an ensembler for vortex core detection.
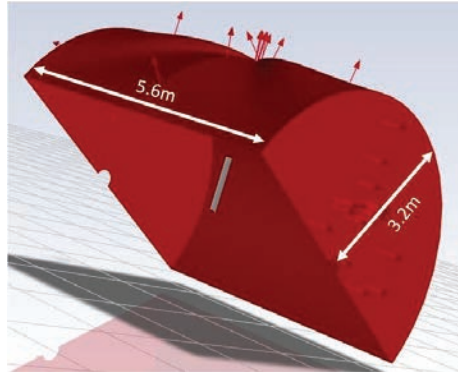
3.1 Simulation

CFD is commonly used to solve and analyse the fluid flow and its effect. It comprises geometry and mesh generation, and fluid modelling. In this research, a rotor blade NACA0012 is selected as a study case for simulation,
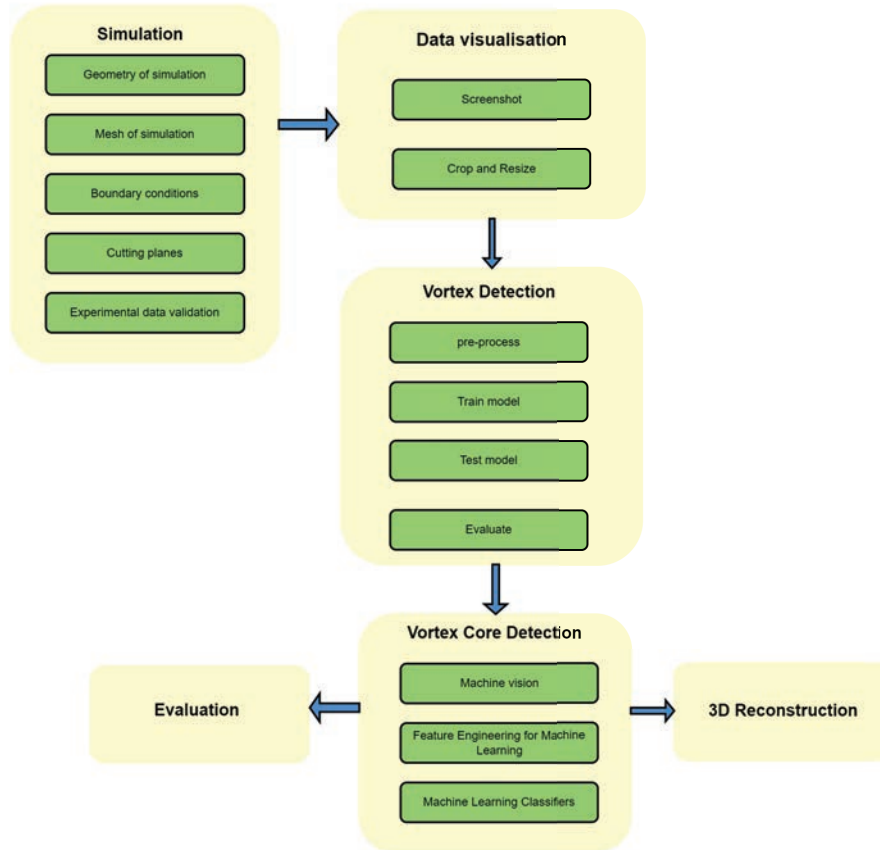
Figure 1 The framework for vortex and vortex core detection.

with the Aspect Ratio (AR) of blade 6 and the radius meter of 1.143 m. Since the case is symmetric, the domain is designed as a half cylinder with a length of 5.6 meters and a radius of 3.2 meters with two periodic planes. The domain dimension is created to have all the results with less space as possible, thus improving the computational efficiency. The geometry of the domain is shown in Figure 2.

The initial mesh is refined locally in the cells where the gradients of the velocity are the lowest, thus having more accurate results near the areas where vortices are likely to occur, as shown in Figure 3. The boundary conditions are modified as per Figure 3, by changing the fluid's rotation-axis direction, rotational velocity, and the rotor blade's angle of attack. The domain is then

**Figure 2**   The simulation domain of a rotor blade NACA0012.
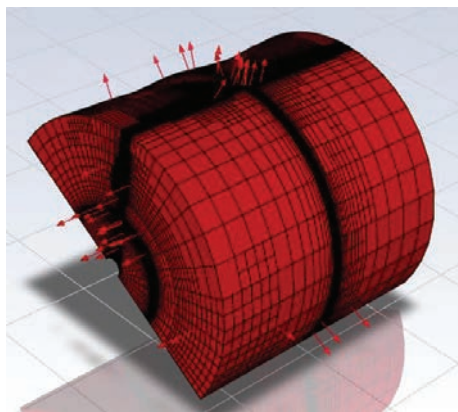
Figure 3   The refined mesh with modified boundary conditions.

sliced into planes to export the data. A total of 158 planes are created, rotated around the z-axis with an angular distance of $1.125°$ between them.

3.2 Data Visualisation

The data with the x, y, and z velocity components is visualised using Paraview, with filters added as a white background. The camera for each plane is set by performing trigonometric calculations from the parameters, with known angles between planes as highlighted in Figure 4(a). The borders of the images are then removed and the images are standardised to a size of 1230px by 580px to reduce the interference and computational cost in

(a) Camera and parameters   (b) A sample image after noise removal
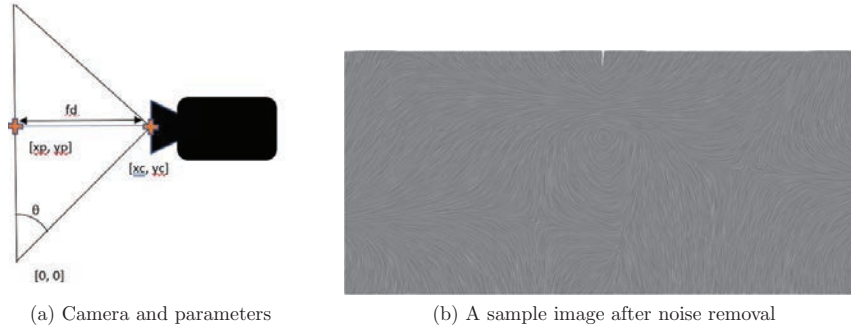
Figure 4   Camera for data visualisation and image sample generated.

subsequent work. Finally, the Gaussian blurring method is implemented to remove noises. A sample image is shown in Figure 4(b).

3.3 Vortex Detection

Vortices are notoriously difficult to predict, and so are their vortex cores. In unsteady flows, the vortex cores as identified by streamlines may not necessarily coincide with the actual vortex cores. However, for steady-state flows, like in the current study, the vortex cores are in agreement with those visualised through streamline plots. Its data pipeline for vortex detection is shown in Figure 5.

Firstly, the images obtained from the simulation are augmented based on Computer Vision techniques which include cropping, panning, rotating and flipping images, adjusting image brightness, and adding noise to images. In total 400 images are generated, which are then labelled for vortex detection using YOLOv5s model. YOLO (You Only Look Once) is well known for its accuracy in real-time object detection. It performs detection in a single pass through an optimized deep Convolutional Neural Network, enabling direct prediction of bounding box coordinates and class probabilities for multiple objects. Here, YOLOv5s is chosen for its high efficiency after comparing their model sizes, loss functions, and accuracy of the YOLO family. The hyperparameters of the model are then configured. For instance, the epochs, learning rate, and optimiser are set to 300, 0.01, and Adam respectively (see detail in the Appendix). Subsequently, the model is trained with weights generated from scratch due to the data specificity and custom object class of vortex and core detection, and evaluated for model performance based on IoU (Intersection Over Union) and confusion matrix metrics.
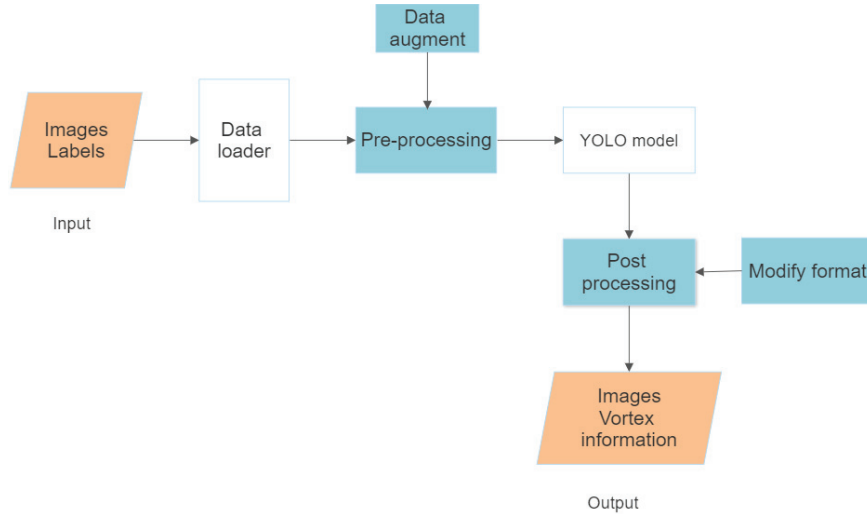
Figure 5   Vortex detection pipeline based on YOLO model.

3.4 Vortex Core Detection

For vortex core detection, this research proposes a computationally light solution by integrating traditional Machine Vision techniques with Machine Learning algorithms, to produce accurate results without the need for a large dataset.
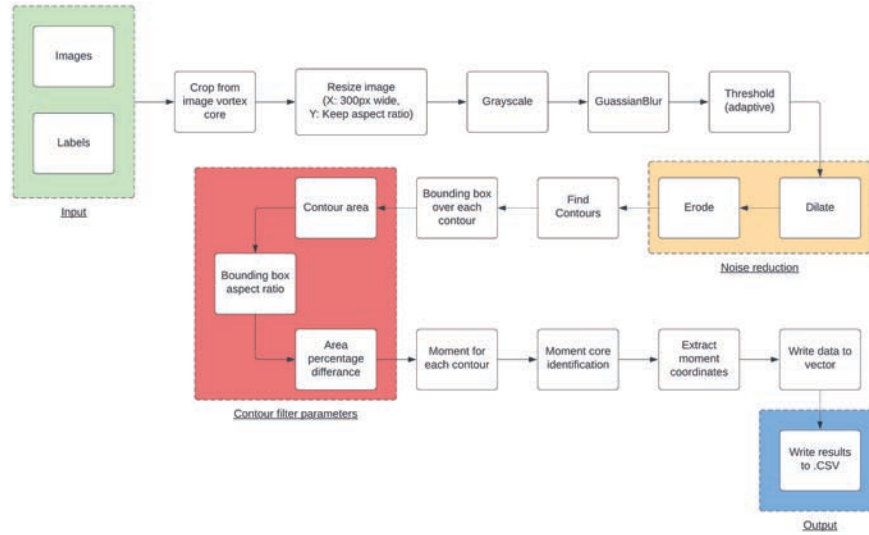
3.4.1 Computer vision pipeline

A Computer Vision based pipeline (as shown in Figure 6) is developed to extract features that best describe the cores while taking computational efficiency and model accuracy into account, before applying Machine Learning to vortex core detection.

As shown in the figure, the labelled images are first supplied to the pipeline with the vortex regions identified in pixel coordinates. The vortex regions are then cropped, separated from the original images, and resized to 300 pixels wide, with the heights calculated below so that the same ARs are kept to ensure that the images are not distorted.

$$New\ Height = Original\ Height * \frac{New\ Width}{Original\ Width}$$

The images are then converted into grayscale, followed by Gaussian blurring to smooth up the image edges, adaptive thresholding to enhance the

Figure 6   Computer Vision based pipeline for feature generation.
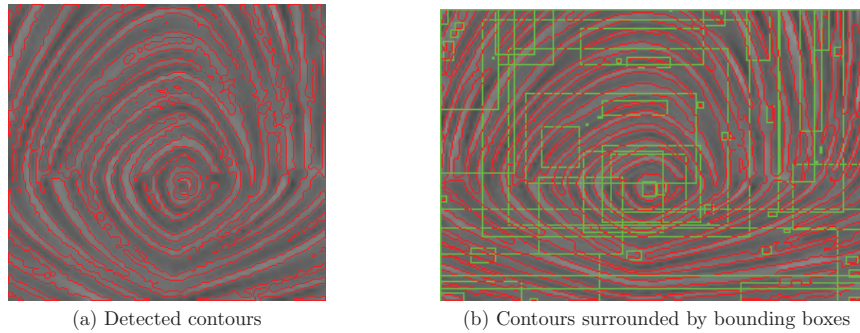

(a) Detected contours   (b) Contours surrounded by bounding boxes

Figure 7   Labelled contours and vortex cores.

contrast between the foreground objects and the image background, and noise reduction to eliminate the unwanted noises by image erosion and dilation (see the resultant sample in the Appendix). Subsequently, contours are detected based on the intensity of the pixels with similar points joined to a continuous path. The bounding boxes surrounding these contours are also generated as shown in Figure 7.

To identify which bounding box/contour parameters best describe a core, a method to filter the redundant information through the data is then implemented with feature filters including Contour Area (CA), Aspect Ratio (AR),
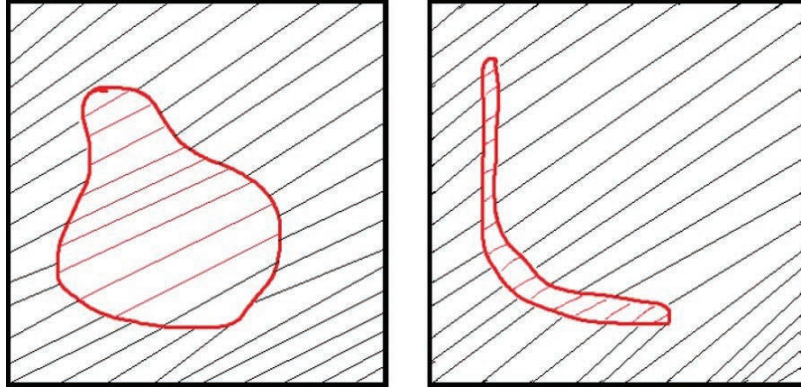
Figure 8 An exemplar illustration of Area Difference (AD).

Area Difference (AD), or Area Percentage Difference (APD), and Moment Centre (MC). For instance, based on observation a vortex core is commonly assigned a contour with a size between large and small extremes as shown in Figure 7. Hence, a threshold size with respect to CA is used to filter out both large and small extremes. As each contour is surrounded by a bounding box, the size of the bounding box can be used to identify the core by analysing the box's AR, i.e., the ratio between an image's width and height. The ARs that best described the vortex cores are then identified with a threshold, beyond which are filtered. AD is the area difference between the bounding box and the contour it surrounds while APD is the percentage of the difference over the bounding box, based on which a tolerance threshold can be defined for locating potential vortex cores. For example, the left figure of Figure 8 is within the tolerance threshold and therefore kept while the right is beyond the threshold and left out.

The sample image through contour filtering is shown in Figure 9(a). The moment (i.e., MC) of each contour is then computed and the one closest to the image's centre is recognised as the vortex core, as shown in Figure 9(b), in which the moment is represented by a blue dot with the contour that it corresponds to highlighted in green. The process culminates in the successful determination of the contour that better describes the vortex core as well as the position of the vortex core.

It is observed that the Machine Vision pipeline produces accurate results subject to the vortex being of a sane shape. While distorted vortices still get misidentified, a Machine Learning ensembler is applied to improve the accuracy of vertex core detection. The parameters used during the filtering
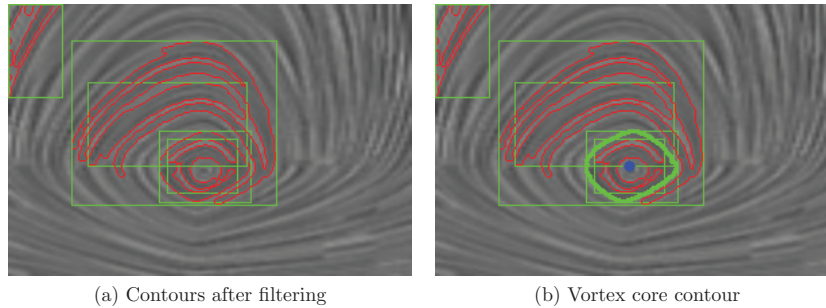
(a) Contours after filtering　　　　　　(b) Vortex core contour

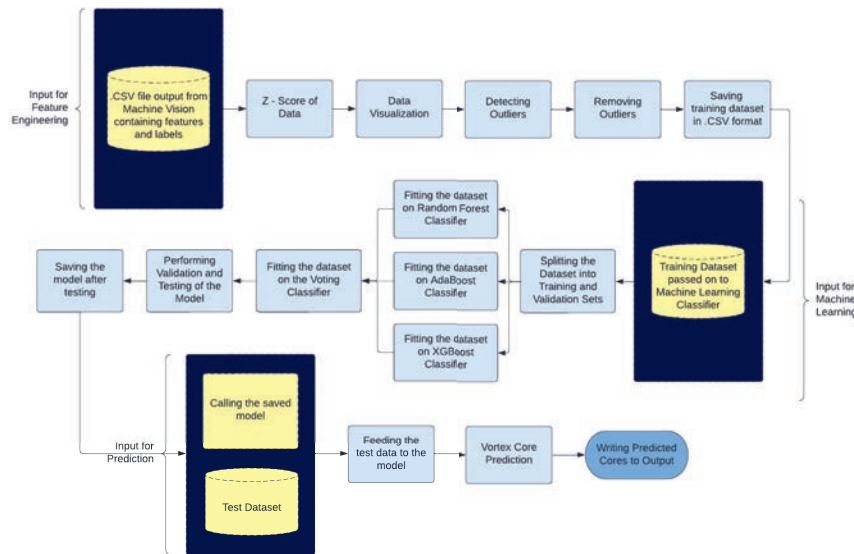**Figure 9**　Labelled contours and vortex cores after filtering.

Figure 10　Machine Learning pipeline for vortex core detection.

process are then saved into a .csv file along with the moment pixel coordinates for a further Machine Learning pipeline.

3.4.2　Machine learning pipeline

The image features in the .csv file are preprocessed with further features extracted before being used by the Machine Learning Classifiers, as displayed in Figure 10.

The features are first normalised using Z-Score as it is an important measure for outlier detection. Then Exploratory Data Analysis (EDA) using

(a) Strip plot

(b) Box plot
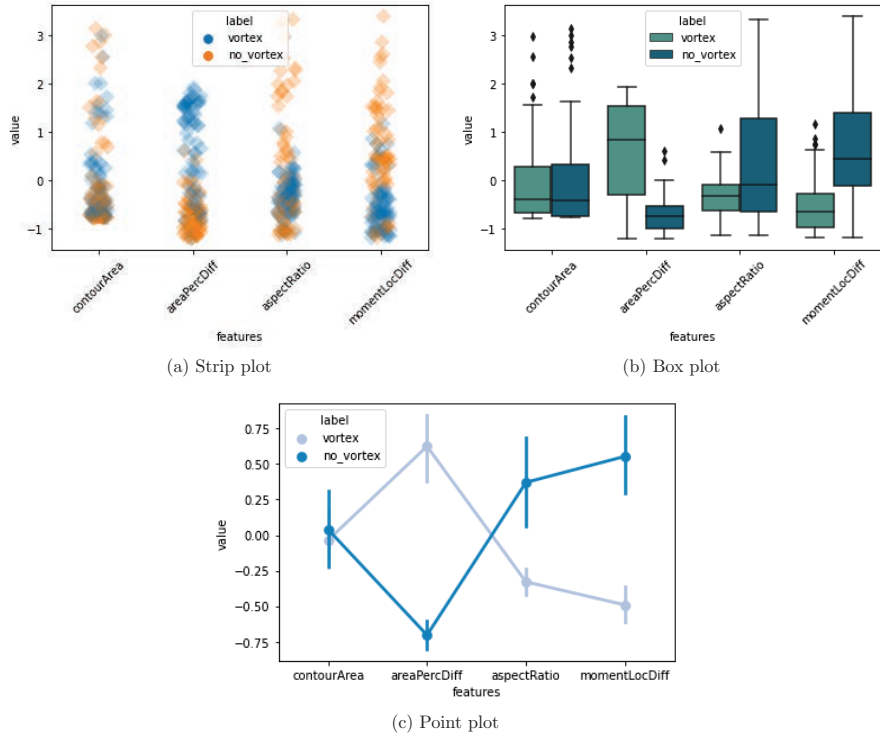
(c) Point plot

Figure 11   Feature distributions using EDA.

strip, box, and point plots and feature heatmap is carried out for potential trend and correlation identification with two categories, i.e., vc (containing vortex cores) and nvc (not containing vortex cores). The EDA is helpful in determining the features that can provide insight into the vortex location. The strip and box plots in Figure 11 visualise the distributions of the vc and nvc values and their relationships by feature. The APD and MC are found to be separable for vc and nvc images.

A point plot shows the estimation of a numerical variable's core propensity and indicates the uncertainty surrounding the estimation. Figure 11(c) visualises where the most vortex and non-vortex values are located by feature based on their means. It confirms the inverse correlation between the two. To further study the correlation between features, a heatmap is implemented in Figure 12. It shows that none of the features are correlated highly enough to be dropped. It is noted that the APD is inversely correlated to the MC as also evidenced by the box plot in Figure 11.
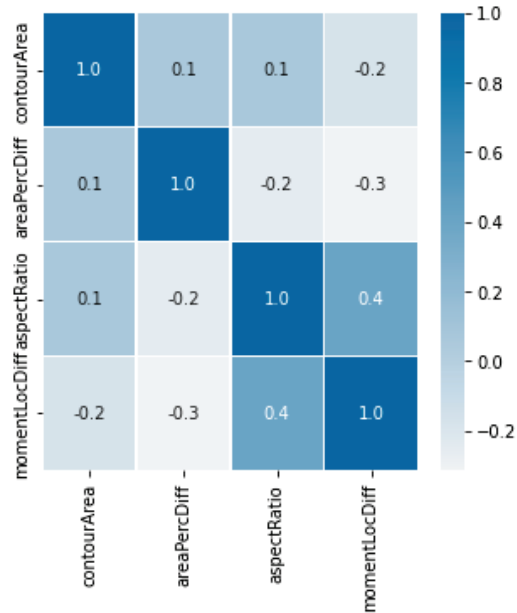
Figure 12   Feature correlation using EDA.

Finally, a hybrid ensemble classifier is implemented in this study, combining both bagging and boosting techniques with the Random Forest, AdaBoost, and XGBoost algorithms (refer to the architecture in the Appendix). Utilizing multiple models, the ensemble classifier is expected to provide more consistent and reliable predictions. The Random Forest algorithm employs bootstrap aggregation (bagging) by training multiple decision trees on different subsets of data and features to aggregate predictions. It excels in handling high-dimensional data and mitigating over-fitting while providing feature importance estimation. Boosting algorithms, on the other hand, sequentially combine weak classifiers by adjusting sample weights to attend to previously misclassified samples, thereby achieving better accuracy compared to bagging. However, they have a tendency to overfit the data. Moreover, XGBoost is a suitable choice for larger or more complex datasets, while AdaBoost is relatively robust against overfitting in datasets with low noise levels. By integrating different models, it is possible to compensate for the limitations of individual models.

The ensemble classifier in this study adopts a hard voting scheme to generate the winner output. Its hyperparameters including $n\_estimators$ (i.e., the number of estimators or decision trees), max_depth (i.e., the maximum

depth for each estimator), $random\_state$ (i.e., the randomness of features) and $learning\_rate$ are tuned based on a grid search method [12]. The confusion matrix and its derived metrics are then applied to evaluate the model performance. Subsequently, the flow of the vortex cores is reconstructed in a 3D shape with the simulation planes.

4 Results

The first step of this study is vortex detection, where a YOLOv5s model is trained using the dataset obtained from a rotor blade NACA0012 simulation. Figure 13 shows the training process of the YOLOv5s model with a sample result of vortex detected based on the augmented dataset. The YOLOv5s
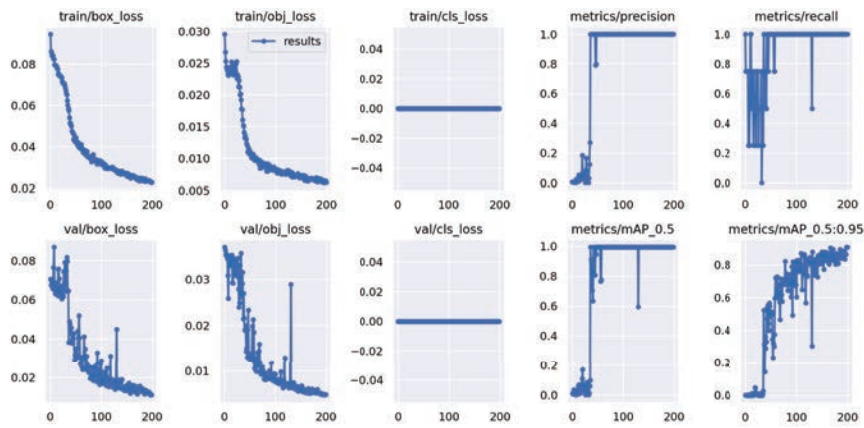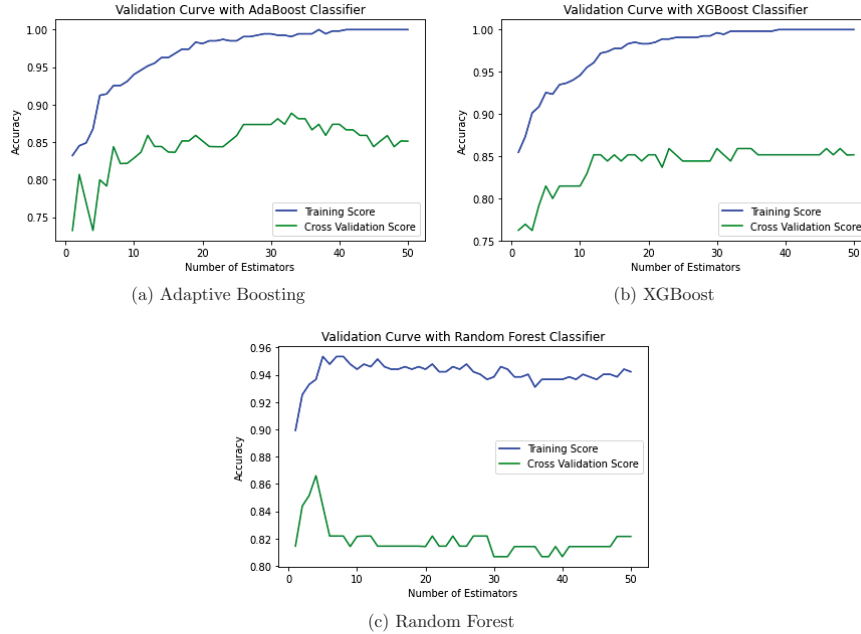

(a) YOLOv5s model training process and metrics
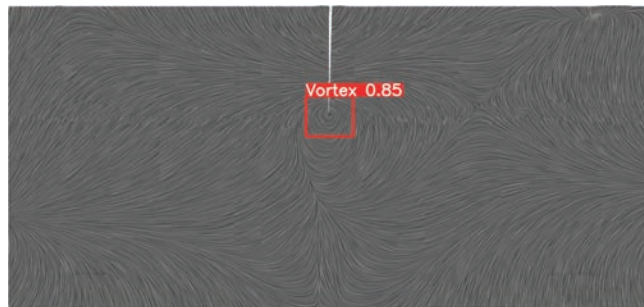
(b) Sample result

Figure 13   Vortex detection sample result using YOLOv5s.

(a) Adaptive Boosting


(b) XGBoost

(c) Random Forest

Figure 14   Training and validation curves of individual classifiers.

training and validation losses (incl. bounding box regression, objectness and classification losses) and metrics (incl. precision, recall and mAP at IoU threshold of 0.5) are shown in figure (a), in which the training is stopped at 200 epochs with minimum classification loss. The figure (b) shows the vortex is correctly identified with a confidence level of 0.85. Through observation of all samples, it is evident that the detection results are accurate with good statistical confidence and no noise distorts the results which therefore can be well applied to the subsequent classification model.

Subseqently, an ensemble classifier is employed to improve the accuracy of vortex core identification by utilizing Computer Vision features extracted from the detected vortex regions. These features are first utilized by three distinct classifiers, i.e., AdaBoost, XGBoost, and Random Forest, to generate classification results individually. Figure 14 displays the performance of the AdaBoost, XGBoost and Random Forest classifiers by the number of estimators. The validation accuracy of AdaBoost stays at 86% above from 25 estimators though falls slightly at the end, the accuracy of XGBoost stabilises at 84% from 25 estimators, and the Random Forest fluctuates between $81\% - 82\%$ from 7 estimators.
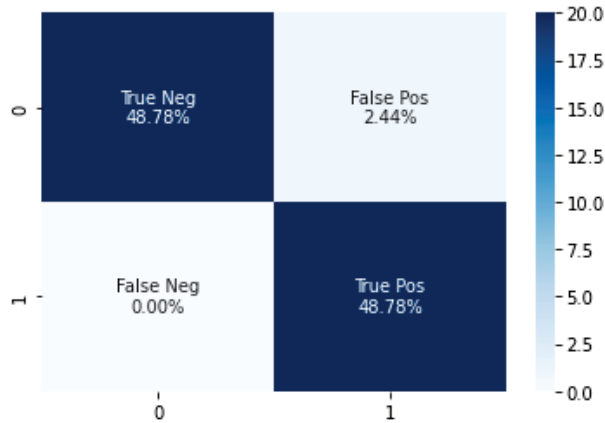
Figure 15 Confusion Matrix based on hard voting ensembler.

Then an ensemble classifier based on the three classifiers is implemented using a majority-take-all voting scheme, through which optimal results are effectively selected from the classifier outcomes. In total 133 images are used by the ensembler in this study, with 93 for training and 40 for testing. The confusion matrix generated by the ensembler is shown in Figure 15.

The figure above shows that the hard voting scheme improves the individual algorithms with an accuracy of 97.56%, in which only one false positive is obtained, without false negative.

5 Discussion

This research develops a vortex and core detection framework by combining traditional Computer Vision and Machine Learning techniques, which is then applied to a rotor blade case. More specifically, the research simulates the wind turbine process, refines its mesh using the lowest velocity gradients, and applies the slicing plane approach to generate vortex images. Then, a vortex detection pipeline based on YOLOv5s, a Feature Engineering and vortex core location pipeline based on Computer Vision, and a Machine Learning pipeline for vortex core detection based on Ensemble Learning are developed. The findings show that the method produces an accuracy score of 97.56% in detecting vortex cores. The results are comparable to those obtained by leading research work such as Abolholl et al. [1] in this area while exhibiting enhanced adaptability and scalability through the utilization of three distinct and comprehensive data processing pipelines.

For small targets such as vortex cores, the Deep Learning approaches such as YOLOs or CNNs alone do not work well. Therefore, this study applies an integrated approach based on multiple technologies including Feature Engineering, Deep Learning, and Ensemble Learning. The labels for vortex core detection are automatically generated by YOLOv5s thus reducing the manual work that other studies (e.g. Biswas et al., [5]) typically rely on. To detect vortex cores, the vortex regions generated by YOLOv5s are passed through Feature Engineering and Machine Learning pipelines for feature extraction and EDA. Then, a hard voting classifier ensembling Random Forest, Adaptive Boosting, and XGBoost algorithms is trained for binary classification.

In feature extraction, the contour approach is selected through trial and error, based on which the ensembler classifies the vortex core images with high accuracy. A number of other features including edge detection, template matching, hough transform, and corner detection are also tested without success. If a vortex is well-shaped and free of distortion, Machine Vision methods alone can produce accurate results, as shown in Figure 9. However, vortices may be irregular, distorted, and noisy. A Machine Vision algorithm combined with ensembler capable of tracking the swirling motion of white lines with noise handling all the way to the vortex core is an option for these vortices, potentially improving the model efficiency and accuracy. Moreover, a further study on hidden image features that are most suitable for vortex core detection is necessary.

This study utilises data augmentation techniques to improve the training of the YOLOv5s model with weights updated from scratch. It shows that the training with a single dataset obtained from the same simulation is effective for vortex and core detection. However, model generalisation often requires training with different datasets, which may negatively impact the model's accuracy. In the study, the simplest YOLOv5s model is selected over some potentially more accurate YOLO models because of its low computational cost. Given more computing power, the accuracy can be improved with more complex YOLO models.

The box plot of the Z-Score values for APD in Figure 16 shows that there is a threshold for the images with and without vortex cores, below which the images are devoid of vortex cores. It is worth investigating whether the variable would be a good detector on vortex cores with a well-tuned threshold, based on a larger dataset with more complex vortices.

Figure 17 shows the visualisation result of all the detected vortex cores in a simulation. The results show the general shape of the flow and are coherent with the expectation.
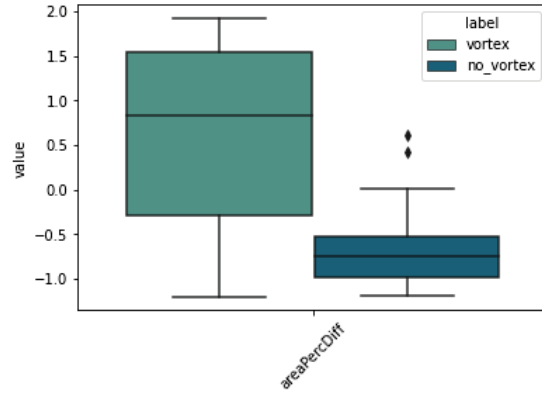
Figure 16 Box plot over the Z-Score values for APD.

(a) Front view

(b) Side view

(c) Top view

(d) Rear view
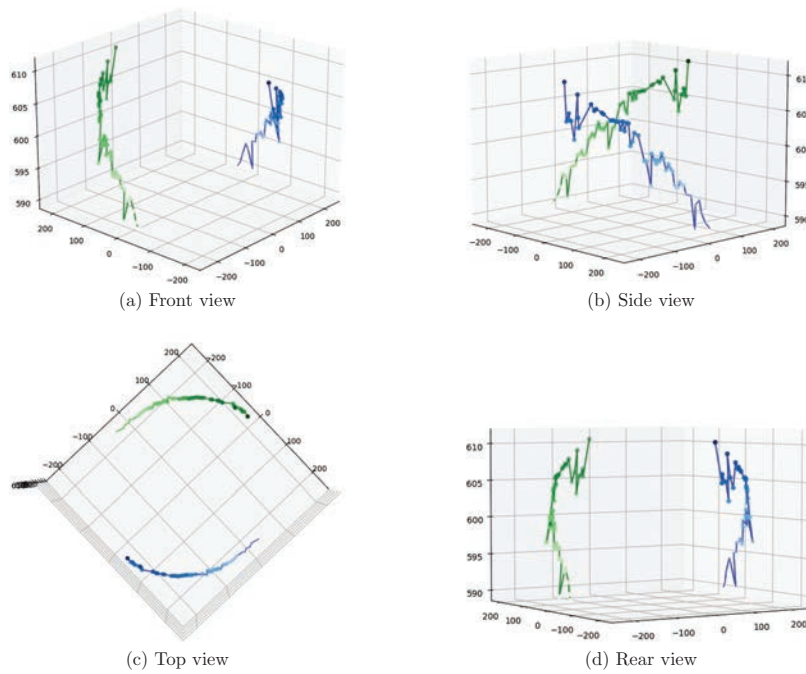
Figure 17 Vortex core flow 3D reconstruction.

This research explores an integrated method for detecting vortices and cores than traditional approaches commonly based on Deep Learning, Machine Learning, or Computer Vision alone. Although high accuracy is achieved by the ensembling approach, the performance of individual

classifiers is impacted by the lack of data diversity (e.g., the vortices of particular types are dominant) and the small size of data samples. It is observed that the proposed solution produces accurate results for the vortices with a circular shape (e.g. oval or spiral) with vortex cores at or near their centres. The images with distorted vortices and cores need to be further tested. Moreover, as the selection of algorithms is arbitrary for ensembler, an uncertainty study is required.

The methodology employed in this study, which incorporates a vortex detection pipeline, Computer Vision-based pipeline, and a Machine Learning pipeline, can be adapted and scaled to address various vortex and core detection scenarios due to the commonality in vortex structural features acquired by YOLOv5s model training and the utilization of Feature Engineering techniques to characterize the vortex cores, evidenced by the distinction in feature distributions between vortex and non-vortex cores. However, further testing and fine-tuning with new datasets are necessary for an optimised solution.

6 Conclusion and Future Work

This research presents a comprehensive framework for detecting vortices and their cores in Computational Fluid Dynamics (CFD) simulations, employing a combination of Computer Vision and Machine Learning techniques. The framework consists of three main components, i.e., a vortex detection pipeline based on YOLOv5s, a Feature Engineering and vortex core detection pipeline based on Computer Vision, and a Machine Learning pipeline utilizing Ensemble Learning for accurate vortex core detection.

The framework is then applied to a rotor blade case, where the fluid simulation vortex images are used to train a YOLOv5 object detector to detect vortices and to extract features by Computer Vision techniques for subsequent vortex core detection by developing a state-of-the-art ensembler. The results demonstrate the effectiveness of the proposed framework in detecting small targets, i.e., vortices and cores, achieving higher accuracy compared to using the YOLO model alone. Furthermore, due to the comprehensive and detailed design, the methodology can be easily adapted and scaled to address various vortex scenarios.

Future work will focus on generalizing the framework by utilizing a larger and more diverse dataset encompassing different types of vortices. Additionally, the framework's uncertainty will be studied, and intelligent thresholds will be explored to enhance the efficiency of vortex and core detection.

A　Appendix
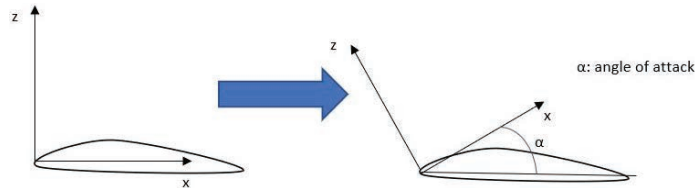
**Figure 1**　Attack angle.

Table 1　Training parameter configuration

Name	Function	Value/Action	Type
weights	initial weights path	None	str
cfg	model.yaml path	ROOT/'models/yolov5s.yaml'	str
data	dataset.yaml path	ROOT/'data/data.yaml'	str
hyp	hyperparameters path	ROOT/'data/hyps/hyp.scratch-low.yaml'	str
epochs	training epoches	300	int
batch-size	total batch size for all GPUs	16	int
imgsz	train, val image size (pixels)	640	int
rect	rectangular training	store_true	–
resume	resume most recent training	False	–
device	cuda device	0,1	–
project	save to project/name	ROOT/'runs/train'	str
name	save to project/name	exp	str

Table 2　Hyper-parameter configuration

Name	Function	Value
lr0	initial learning rate	0.01
lrf	final OneCycleLR	0.01
momentum	0.937	SGD momentum/Adam betal
weight_dacay	optimizer weight decay	0.0005
warmup_epochs	warmop epochs	3.0
warmup_momentum	warmup initial momentum	0.8
warmup_bias_lr	warmup initial bias lr	0.1
box	box loss gain	0.05
cls	cls loss gain	0.5
cls_pw	cls BCELoss positive_weight	1.0
iou_t	IoU training	threshold
anchor_t	anchor-multiple threshold	4.0

(a) Vortex core　　　　　　(b) Noise reduction
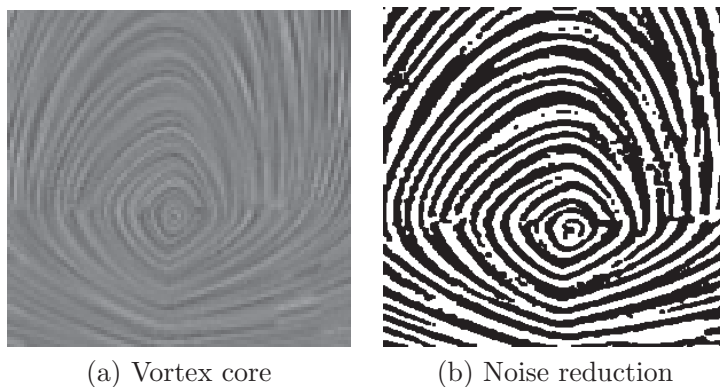
Figure 2　Vortex region generated through noise reduction process.
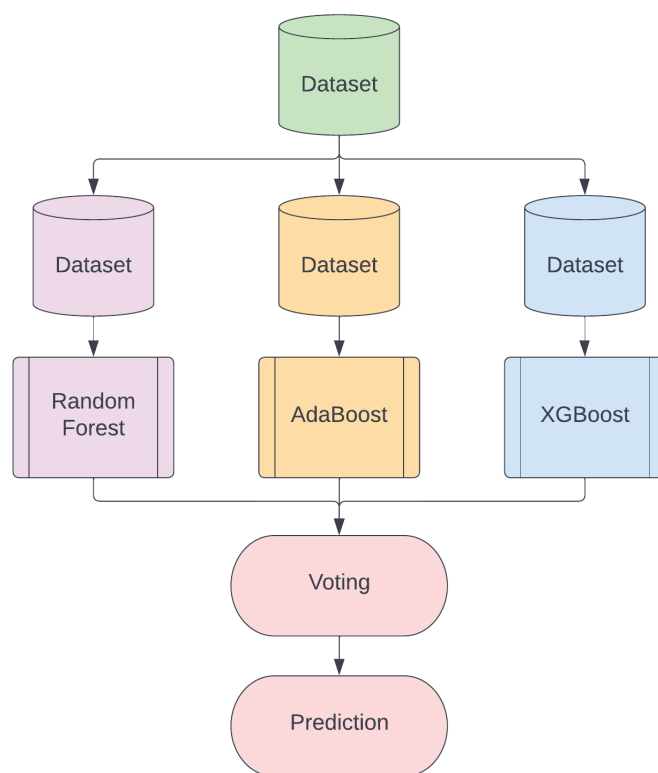
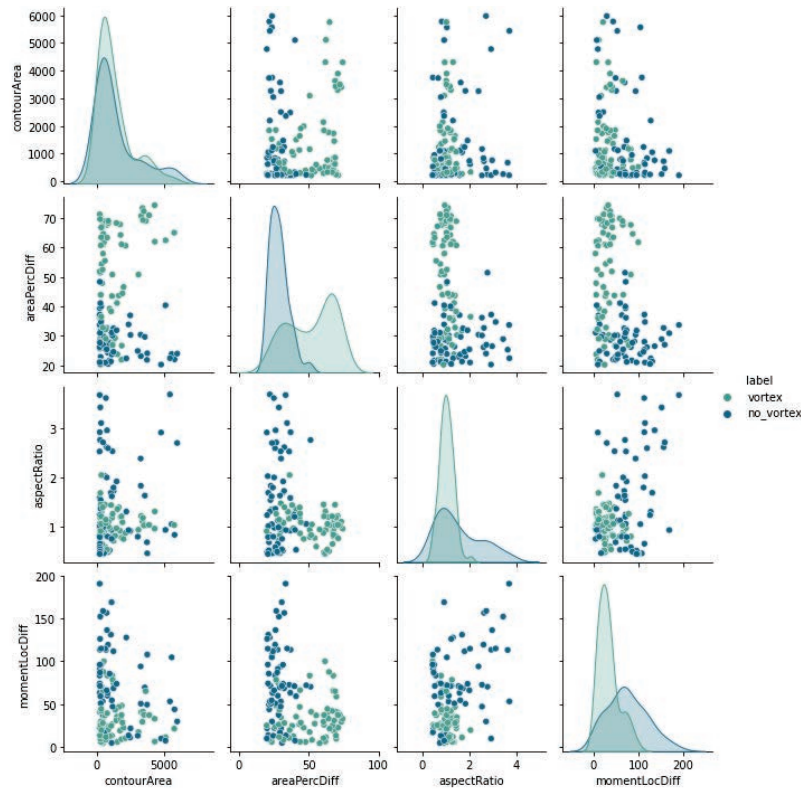**Figure 3**　The ensemble classifier based on hard voting.

Figure 4   Feature pair plot.

References

[1] Hazem Abolholl, Tom-Robin Teschner, and Irene Moulitsas. "A hybrid computer vision and machine learning approach for robust vortex core detection in fluid mechanics applications". In: Apr. 2022.

[2] Hazem Ashor Amran Abolholl, Tom-Robin Teschner, and Irene Moulitsas. "Surface Line Integral Convolution-Based Vortex Detection Using Computer Vision". *Journal of Computing and Information Science in Engineering* 23.5 (2023). 051002. ISSN: 1530-9827. DOI: 10.1115/1. 4056660. URL: https://doi.org/10.1115/1.4056660.

[3] Jennifer Abras and Nathan S Hariharan. "Application of Machine Learning to Automate Vortex Core Extraction Computations in Hovering Rotor Wakes". In: *AIAA AVIATION 2021 FORUM*. 2021, p. 2595.

[4] Syeda Sarah Azmi and Shwetha Baliga. "An Overview of Boosting Decision Tree Algorithms utilizing AdaBoost and XGBoost Boosting Strategies". In: *Int. Res. J. Eng. Technol.* 7.05 (2020).

[5] Ayan Biswas et al. "An uncertainty-driven approach to vortex analysis using oracle consensus and spatial proximity". In: *2015 IEEE Pacific Visualization Symposium (PacificVis)*. 2015, pp. 223–230. DOI: 10.110 9/PACIFICVIS.2015.7156381.

[6] Dietterich. "An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization". In: *Machine Learning* 40.139–157 (2000).

[7] Ming Jiang, Raghu Machiraju, and David Thompson. "Detection and visualization of vortices". In: *The visualization handbook* 295 (2005).

[8] U Karthik Kumar, MB Sai Nikhil, and K Sumangali. "Prediction of breast cancer using voting classifier technique". In: *2017 IEEE international conference on smart technologies and management for computing, communication, controls, energy and materials (ICSTM)*. IEEE. 2017, pp. 108–114.

[9] C Levit and T Lasinski. "A tool for visualizing the topology of three-dimensional vector fields". In: *Proc. Visualization'91*. Citeseer. pp. 33–40.

[10] Yanyang Luo et al. "CNN-based blade tip vortex region detection in flow field". In: *Eleventh International Conference on Graphics and Image Processing (ICGIP 2019)*. Vol. 11373. International Society for Optics and Photonics. 2020, 113730P.

[11] Maciej Majchrzak, Mateusz Jakubowski, and Roman Starosta. "AI-based method of vortex core tracking as an alternative for Lambda2". In: *Vibrations in Physical Systems* 31.3 (2020).

[12] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[13] Dymitr Ruta and Bogdan Gabrys. "Classifier selection for majority voting". In: *Information fusion* 6.1 (2005), pp. 63–81.

[14] Omer Sagi and Lior Rokach. "Ensemble learning: A survey". In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 8.4 (2018), e1249.

[15] David Sujudi and Robert Haimes. "Identification of swirling flow in 3-D vector fields". In: *12th Computational fluid dynamics conference*. 1995, p. 1715.

[16] Jun Wang et al. "A Vortex Identification Method Based on Extreme Learning Machine". In: *International Journal of Aerospace Engineering* 2020 (2020).

[17] Shiying Xiong et al. "Neural vortex method: from finite lagrangian particles to infinite dimensional eulerian dynamics". In: *Computers & Fluids* 258 (2023), p. 105811.

Biographies
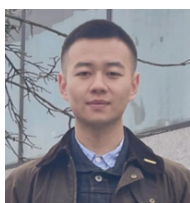
Zhenguo Xu received a bachelor's degree in Information management and Information Systems from Shenyang Aerospace University in 2018, and a master's degree in Computational and Software Techniques in Engineering from Cranfield University in 2022. He is currently studying for his Ph.D. at the Centre for Computational Engineering Sciences. His research areas include complex networks and Deep Learning.

Zhenguo Xu, Ayush Maria, Kahina Chelli, Thibaut Dumouchel De Premare, Xabadin Bilbao, Christopher Petit and Robert Zoumboulis-Airey are MSc graduates (2022) at Centre for Computational Engineering Sciences, School of Aerospace, Transport and Manufacturing, Cranfield University.

Irene Moulitsas is a Senior Lecturer at Cranfield University and a member of the Digital Aviation Research and Technology Centre (DARTeC) with

substantial experience in large-scale Computing, Algorithms, Simulation and Modelling. She is the Course Director for MSc Computational Software Techniques in Engineering. Irene's research has been funded through national and regional bodies in USA and EU and has focused on developing novel algorithms for enabling the efficient execution of large scientific computations on serial and parallel processing platforms. She has developed highly efficient algorithms and software that are publicly available for use by universities, research laboratories and companies. In 2021 Irene was selected as a finalist for the Top 50 Women in Engineering in the UK award.


Tom Teschner is a lecturer in Computational Fluid Dynamics with a background in aerospace and software engineering. His research focuses on pressure-velocity coupling algorithms for incompressible flows and applied aerodynamics for aerospace and automotive applications.

Seemal Asif is a skilled CU Control Lead overseeing an ATI-funded Airbus project. With expertise in academia and industry, she prioritizes human involvement in automation. She leads a research group (IFRA) specializing in industrial robotics, focusing on areas such as robotic path

correction, imitation learning, and coordinated swarm communication. As a Co-Investigator on EPSRC-funded projects worth millions of pounds, she plays a crucial role in developing frameworks and integrating the human factor in responsive manufacturing and industrial human-robot collaboration. Seemal's dedication to solving real-world problems while considering the well-being of operators demonstrates her commitment to advancing automation technologies in a safe and efficient manner.
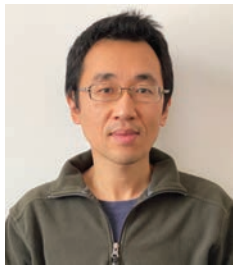
Jun Li is a lecturer in Mathematics and Computer Science at Cranfield University, UK. He has a BSc, MSc and PhD in Computer Science, Software Engineering and Artificial Intelligence respectively. Before joining Cranfield University, Dr Jun Li taught at the University of Wolverhampton and LondonMet University for five years and also worked as a Research Associate at the University of Cambridge and the University of Oxford for six years. His teaching and research expertise are in the areas of Machine Learning, Data Analytics, Computer Science and Mathematical Modelling applied to various domains.