

A HYBRID COMPUTER VISION AND MACHINE LEARNING APPROACH FOR ROBUST VORTEX CORE DETECTION IN FLUID MECHANICS APPLICATIONS

Hazem Ashor Amran Abolholl¹, Tom-Robin Teschner^{1,*}, Irene Moulitsas¹

¹Cranfield University, College Rd, Cranfield, Wharley End, Bedford MK43 0AL

ABSTRACT

Vortex core detection remains an unsolved problem in the field of experimental and computational fluid dynamics. Available methods such as the Q , delta, and swirling-strength criterion are based on a decomposed velocity gradient tensor but detect spurious vortices (false positives and false negatives), making these methods less robust. To overcome this, we propose a new hybrid machine learning approach in which we use a convolutional neural network to detect vortex regions within surface streamline plots and an additional deep neural network to detect vortex cores within identified vortex regions. Furthermore, we propose an automatic labelling approach based on K-means clustering to pre-process our input images. We show results for two classical test cases in fluid mechanics; the Taylor-Green vortex problem and two rotating blades. We show that our hybrid approach is up to 2.6 times faster than a pure deep neural network-based approach and furthermore show that our automatic K-means clustering labelling approach achieves within 0.45% mean square error of the more labour-intensive, manual labelling approach. At the same time, using a sufficient number of samples, we show that we are able to reduce false positives and negatives entirely and thus show that our hybrid machine learning approach is a viable alternative to currently used vortex detection tools in fluid mechanics applications.

Keywords: machine learning; computer vision; vortex core detection; fluid mechanics

2010 MSC : 68T45, 76D17, 76F65, 76M27

1. INTRODUCTION

It is widely considered that vortex dynamics play an essential role in determining the behaviour of fluid flows in practical applications. For instance, identifying the locations of vortices on the aircraft objects consigning to reduced drag and lift. It is also beneficial to know the exact location of vortex cores when automated mesh refinement in computational fluid dynamics simulations (CFD). Therefore, during the last decades, researchers have intensively studied the identification of vortices in flow fields.

Two types of algorithms are commonly used to detect vortices, local and global. The local method is based on the decomposition of the local velocity gradient tensor such as Q criterion [1], delta criterion [2], Lambda 2 criterion [3], and swirling-strength criterion [4]. These techniques could acquire reasonable estimates, yet, at the same time, these techniques generate a substantial number of false positives and negatives. To control the number of these false positives and negatives, the user should set appropriate parameters which lead to poor robustness. The Global vortex detection methods are typically accomplished through streamlines to detect vortex regions. Generally, global methods are more robust than local methods, but they use neighbouring cells to identify vortical flows, which adds to their computational time. On the other hand, local methods require users' input who have specific knowledge about the field, making them inappropriate to detect vortex cores automatically. Due to this, both local and global detection methods have weaknesses and cannot provide fully robust and reliable results.

To address this problem, we present a hybrid computer vision and artificial neural network (ANN) machine learning approach to detect vortex cores in this study. Initially, we employed computer vision to identify the areas of vortical structures [5]; then, we applied ANN to identify the vortex core within the vortex region. You Only Look Once (YOLO) algorithm [6] has been used in the computer vision part to identify the areas where the vortex cores are located. YOLO applies a neural network to the whole image at once rather than making predictions on individual parts within the image. Then, ANN is added on top, where this hybrid approach of computer vision using YOLO and machine learning using ANN will add new dimensional. ANN is a set of features that can be used not only to identify the vortices in general but also to detect the vortex core, which is an important application in fluid mechanics. The benefit of tracking these vortices is to know how the flow is developing. The ANN algorithm is performed using Keras, which is a deep learning Application Programming Interface (API) acting as an interface for TensorFlow [7]. TensorFlow is an open-source written in Python for machine learning.

*Corresponding author: tom.teschner@cranfield.ac.uk

Documentation for asmeconf.cIs: Version 1.34, February 16, 2024.

2. LITERATURE REVIEW

Computed fluid dynamics has been applied to diverse areas such as medical [8, 9], aircraft navigation during nose-ups and rolls of jets [10], as well as industrial applications [11]. CFD modelling is used to simulate turbulent flow with vortex structures of different scales of motion [12]. The book "Vortex Dynamics," by Saffman [13], defines a vortex as "a finite volume of vorticity immersed in irrotational fluid". This definition, however, does not satisfy since it requires "that the fluid is ideal for making strict sense because viscosity diffuses vorticity" [13]. Lugt [14] described a vortex: "A vortex is the rotating motion of a multitude of material particles around a common centre". This definition does not lend itself to practical implementation in a detection algorithm [15]. Another definition from Robinson [16] is that "a vortex exists when instantaneous streamlines mapped onto a plane normal to the vortex core exhibit a roughly circular or spiral pattern when viewed from a reference frame moving with the centre of the vortex". This definition is self-referential because, in order to identify the vortex core, we would need prior knowledge of its location, orientation, and motion.

Although no formal definition exists, identifying vortices using the methods currently available has been extensively documented in recent years. The two common approaches to detecting vortices are local and global algorithms [17–19]. Local methods rely on the velocity gradient tensor. These methods produce quick results and are relatively easy to implement. Nevertheless, some parameters need to be carefully selected to guarantee robust results, thus increasing the time required to identify vortex cores. Global vortex identification methods are generally based on streamlines to identify vortex regions. Some of the most widely used global methods are the winding angle [17], the instantaneous vorticity deviation (IVD) [18], and the elliptic objective Eulerian coherent structures (OECS) [19]. Generally, global methods are more precise than local methods due to their use of neighbouring cells to identify vortical flows, but they require more computational time than local methods [20]. Moreover, global methods require user input, so they are unsuitable for detecting vortex cores automatically. Consequently, both local and global methods have their limitations and cannot obtain completely robust and reliable detection results.

In this study, we employed computer vision and machine learning approaches to detect vortex cores in a flow field in order to overcome existing problems and eliminate the current limitations of traditional local and global vortex identification methods. In our approach, we infer vortex locations directly from images and the data to generate these images, rather than relying on surrogate quantities such as the velocity gradient tensor, as used by local and global methods. For vortex extraction, several computer vision approaches have emerged in recent years. In computer vision, a wide range of techniques is conceived for extracting, tracking, and identifying features from images [21, 22]. Scientists rely on visualisation to understand, analyse, and identify physical phenomena or mathematical models, so tracking features in their fields differs from most computer vision applications. A computer vision technique was recently developed to detect vortical structures in the flow field in order to obtain highly robust and reliable measurements. In a computational research study using deep neural

networks, the reconstruction of velocity and pressure fields in turbulent flow with high resolution has been tested, as well as identifying the characteristic features of the turbulent flow, using computer vision [23]. By using a convolutional neural network (CNN), Franz et al. [24] detected and tracked ocean eddies and provided the labelled training data based on local vortex identification. From sea surface height maps, Lguensat et al. [25] used a deep neural network (DNN) to detect ocean eddies by using the entire visualisation images of the flow field. In a study by Deng et al. [26], they developed a vortex identification method based on a CNN, using the global instantaneous vorticity deviation method along with a local method to determine each point's label. The local vortex detection algorithms are used to label data sets that are likely to contain spurious vortices, and therefore the network can be trained using false positives and false negatives.

In recent years, notable advancements have been observed in machine learning methodologies, capable of autonomously determining essential attributes from extensive datasets. These achievements have been particularly apparent in diverse domains, such as fluid dynamics. This success can be attributed to progress in algorithms, with a focal point on deep learning [27], facilitated by the contemporary strides in computational power. In these instances, supervised machine learning has played a pivotal role by approximating the intricate connections between input and target data, leveraging its remarkable universal approximation capabilities [28, 29]. Lee et al. [30], support vector regression is employed within the field of fluid dynamics to differentiate between diverse flow conditions and dynamic regimes. Support vector machines (SVMs) have established themselves as a classical machine learning algorithm for addressing classification challenges. This supervised learning approach is grounded in statistical learning theory, encompassing the principle of structural risk minimization. These endeavours have given rise to a substantial body of research, featuring an array of machine-learning methods, all accomplished within a relatively brief span. Machine learning methodologies were additionally devised for the purpose of vortex classification and identification, as evidenced by studies conducted by Ribeiro and Franck [31], and Deng et al. [26].

We propose in this study an alternative way to use CNN to track and detect the region of vortical structures on streamline plots by replacing local detection methods for labelling vortices manually and automatically by employing computer vision and unsupervised learning algorithms (K-Means Clustering). This approach allows us to obtain just the region of vortices and eliminate false positives and negatives. As a result, CNNs will be more accurate for identifying vortex areas and the training time of the machine learning algorithm is also reduced to detect vortex cores. The following section describes the machine-learning approach to detecting vortex cores.

3. METHODOLOGY

3.1 Streamline detection using the line integral convolution algorithm (LIC)

In order to prepare an image data set for a computer vision framework, we use the LIC algorithm as a vector field technique. As proposed by Cabral and Leedom [32], LIC is used to extract

the velocity vector from each pixel and compute its local streamlines from that vector. This numerical algorithm integrates along streamlines, but the speed depends on the vector field at that point, so in regions with a fast flow, the lines will be long, and vice versa. Streamline plots are generated by covering a surface with a white noise image value at the resolution required for the plot. It then computes the following integral:

$$C(\mathbf{r}) = \int_{-L/2}^{L/2} k(s)N(\sigma(s))ds \quad (1)$$

where $C(r)$ represents the colour value at location r , s being the arc length along the streamline, $k(s)$ is an appropriately chosen kernel function to limit colouring to local space and $N(\sigma(s))$ containing the grey-scale image. Integration over a distance L is performed along the streamline ds . Fig 1 shows an example of the output produced by the LIC algorithm. In contrast to the traditional flow line method, LIC does not produce line primitives instead, it generates a texture whose pixels are coherent along the flow direction. This will make it very easy to visualise the flow direction everywhere in the domain. The LIC module is



FIGURE 1: VECTOR FIELD VISUALIZATION (LINE INTEGRAL CONVOLUTION).

available in ParaView, where we generate our streamline plots. These images are then further pre-processed using the Torchvision module — which is part of PyTorch Python — in order to make the training more robust by rotating, cropping, and flipping them horizontally or vertically flipping data augmentation. In order to train the YOLO-supervised algorithm, we manually label vortex regions on the LIC images. Although this is a lengthy process, it has to be carried out just once to be able to apply it to different fluid problems with confidence.

3.2 YOLO

YOLO algorithm [6] has gained attraction in the past few years, based on advanced research achievements in the field of computer vision, it has achieved a good balance between speed and accuracy [33, 34] for object detection. YOLO is a regression problem that passes the entire image into a Neural Network. In Fig 2, the YOLO framework is illustrated. In this method, the input image is divided into a $S \times S$ grid cells and perform detections in each grid cell. A grid cell is responsible for detecting an object

whose centre falls into it. Each cell predicts B bounding boxes along with the confidence scores for these boxes. Based on the confidence scores, we can determine whether an object exists in a grid cell. The confidence is formulated as follows:

$$Confidence = P_r(object) \times IoU_{pred}^{truth} \quad (2)$$

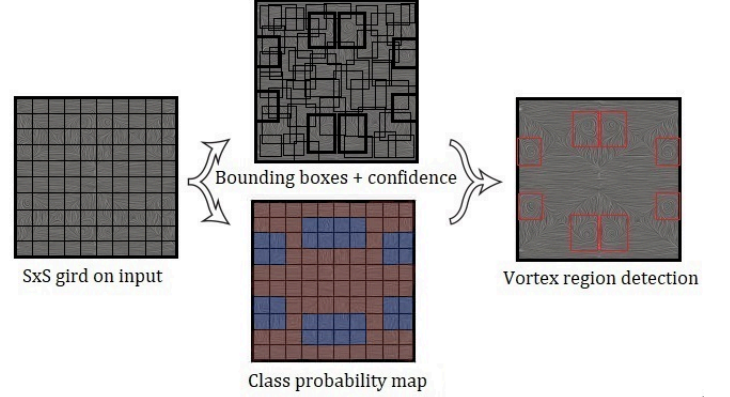


FIGURE 2: YOLO MODEL DETECTION.

Where $P_r(Object) \in [0, 1]$ and IoU is the Intersection over Union between the predicted bounding box and the ground truth labelling. In the case where there is no object in that cell, the confidence score equals zero, whereas if there is an object, the confidence score is equal to the IoU . Each bounding box consists of $(5+C)$ values to predict the object as follows: x , y , w , h , confidence and C class probabilities. (x, y) represent the centre coordinates of the box, and (w, h) represent the width and height of the box, respectively.

In this study, we utilised YOLO version 3 [35] for identifying the region of vortex cores, improving detection accuracy and reducing false positives and false negatives. The backbone network of YOLOv3 is called Darknet-53, which has 53 CNN layers, to extract feature images from the input. A schematic of the YOLOv3 architecture is illustrated in Fig 3. It consists of residual blocks, skip connections, and up-sampling layers, where each convolutional layer is followed by batch normalization layers and leaky ReLU activation functions. In YOLOv3, objects are detected at three different scales in the network, and the feature maps are extracted on layers 82, 94, and 106. The first scale is stride 32 which divides the 416 pixels' input image into 32 to produce an output size of 13 by 13, the second scale is stride 16 which produces an output size of 26 by 26, and the last scale is stride 8 which produces an output size 52 by 52. Therefore, strides 8, 16 and 32 are responsible for detecting small, medium and large vortices in our LIC images, respectively. The ability to detect features at three different scales makes YOLOv3 a better alternative to YOLOv1 and YOLOv2, especially when it comes to detecting smaller vortices.

3.3 Unsupervised K-Means Clustering Algorithm

In data science, clustering is a method for finding clusters in a data set that are characterised by the greatest similarity within a cluster and by the greatest dissimilarity between clusters [37].

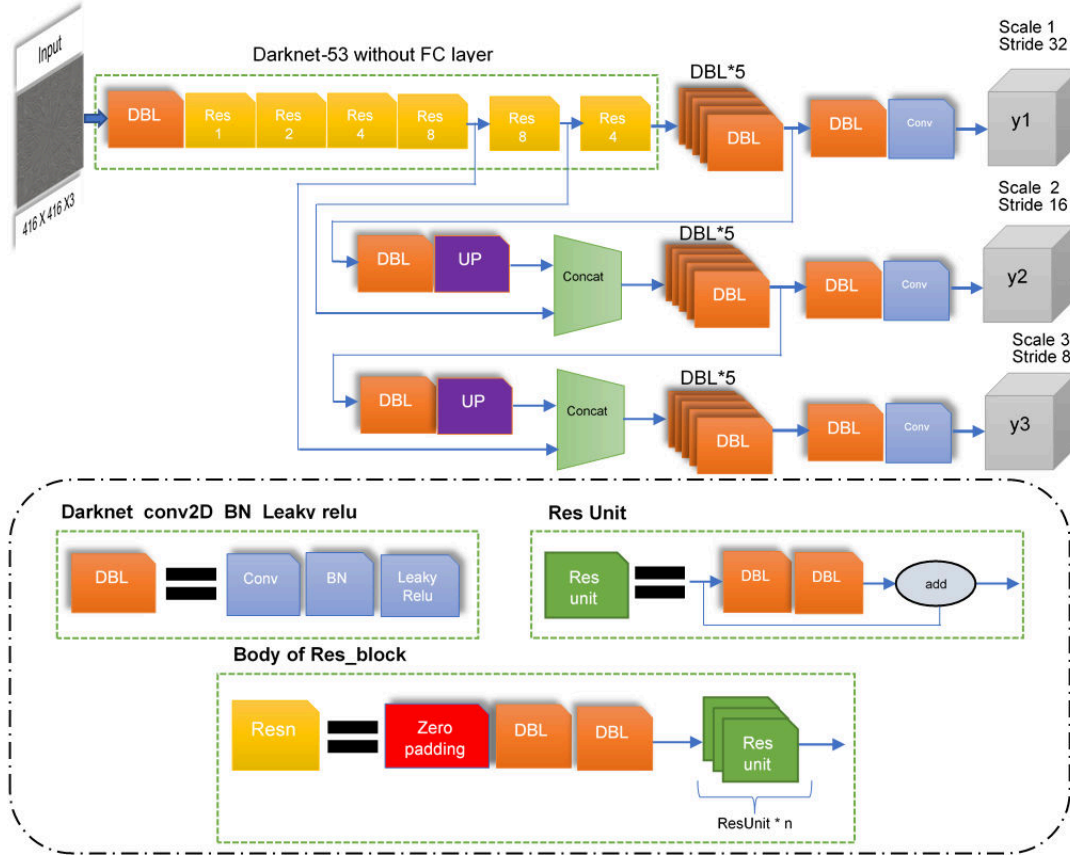


FIGURE 3: YOLOV3 ARCHITECTURE WITH INPUT IMAGE SIZE 416×416 AND 3 TYPES OF SCALES (13×13, 26×26, AND 52×52) AS OUTPUT; DARKNET-CONV2D-BN-LEAKY ("DBL" FOR SHORT) IS THE MAIN COMPONENT OF YOLOV3 WHICH CONTAINS ONE CONVOLUTION LAYER, ONE BATCH NORMALISATION LAYER, AND ONE LEAKY RELU LAYER.; RESIDUAL-LIKE UNIT ("RESUNIT" FOR SHORT) IS TWO "DBL" STRUCTURES FOLLOWING ONE "ADD" LAYER; SEVERAL "RESUNIT" WITH ONE ZERO-PADDING LAYER AND "DBL" STRUCTURE FORWARD GENERATES A RESIDUAL-LIKE BLOCK, "RESBLOCK" FOR SHORT, WHICH IS THE MODULE ELEMENT OF DARKNET-53. FIGURE ADAPTED FROM WANG ET AL. [36]

Let us consider a data set $X = x_1, x_2, \dots, x_n$, where each point is a d-dimensional real vector. As the first step in the algorithm, initial estimates for the K centroids are generated by random selection. Using the squared Euclidean distance, each data point x is assigned to its nearest centroid. Let $M = m_1, m_2, \dots, m_c$ where m_c is cluster centres. The data point x is assigned to a cluster based on the following:

$$J(w, M) = \sum_{k=1}^c \sum_{i=1}^n w_{ki} \|x_i - m_k\|^2 \quad (3)$$

Where $w_{ki} = 1$ for data point x_i when it belongs to cluster k; otherwise, $w_{ki}=0$. After that, the centroids are recalculated to update them by taking the mean of all data points assigned to that centroid's cluster as follows:

$$m_k = \frac{\sum_{i=1}^n w_{ki} x_i}{\sum_{i=1}^n w_{ki}} \quad (4)$$

$$\Rightarrow w_{ki} = \begin{cases} 1, & \text{if } k = \operatorname{argmin}_j \|x^i - m_j\| \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

As an iterative process, the K-means algorithm repeats each step until no data points change clusters, the distances are minimised, or there is some maximum number of iterations.

In this study, the flow field files were automatically labelled for training our model with computer vision by identifying close regions around vortex cores, and clusters of these areas were identified using a K-means clustering algorithm. After applying K-Means to cluster pixels of the vortex core regions from each LIC image into groups of similar pixels, then, we obtain the centre of a cluster which is the average of the pixels in that cluster. It is also called a "centroid" which represents the vortex core.

3.4 Deep learning model for vortex core identification

Artificial neural networks (ANNs) are one of the subsets of machine learning. It consists of a set of algorithms and a set of rules that are based on the biological neural networks that are found in our brains for the purposes of classification and regression [38]. In neural networks, input layers are related to outputs using activation functions. Non-linear neuron theories can be expressed as follows:

$$h_{w,n}(x) = f \left(\sum_{i=1}^{PL_n} w_i x_i + b \right) \quad (6)$$

Where b is the bias, w_i 's are weights of the network, PL_n is

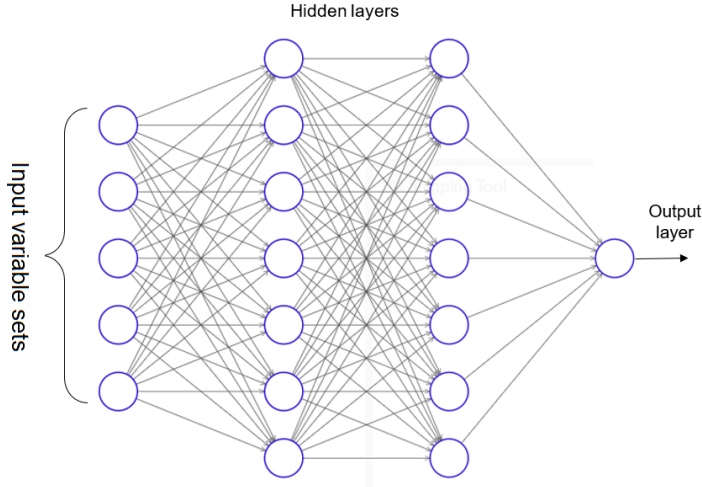


FIGURE 4: NEURAL NETWORK DIAGRAM FOR A FEED-FORWARD NN WITH INPUTS LAYERS, HIDDEN LAYERS, AND OUTPUT.

the number of neurons at layer n , and f is called the activation function which can be chosen as a Sigmoid (logistic function), TanH (hyperbolic tangent), ReLu (rectified linear unit), or Soft-max [39]. A standard feed-forward deep neural network is shown in Fig 4. The neural network operates by combining data sources (inputs) and transforming them through nonlinear activation functions. The procedure is repeated once for each hidden layer in the network until the output layer is reached. The objective of this study is not only to identify the areas of vortices but also to detect the vortex cores. Therefore, after identifying the region of vortices by using YOLOv3 [5], we just obtain the numerical data of flow field files within these regions. This pivotal step constitutes a profound reduction of the numerical dataset, achieved by selectively capturing the values that are in immediate proximity to the vortex cores. In addition, we also fetch whole numerical data of flow field files for the LIC image for training pure ANN to compare its results with the hybrid approach. Then the data set has been labelled manually at first and automatically by using computer vision and k-means clustering. By utilising computer vision, we have the capability to determine the number of clusters by counting the bounding boxes encircling the vortex cores. The data set is split into 70% training data set, 20% validation data set to overcome overfitting, and 10% test data set to test the model. The features of the data set for the single approach (pure ANN) and our hybrid approach (YOLO and ANN) are divided into five variable sets as follows:

1. x-velocity, y-velocity, w-velocity, velocity magnitude and pressure
2. Set 1 + velocity gradient tensor and pressure gradient

3. Set 1 + u-vorticity, v-vorticity, w-vorticity and vorticity magnitude
4. Set 2 + u-vorticity, v-vorticity, w-vorticity and vorticity magnitude
5. Set 4 + dynamic pressure, normalised Q-criterion and absolute Q-criterion

In this study, we use TensorFlow and the Keras API, open-source Python libraries to train our model.

4. PROBLEM DESCRIPTION

In this study, we use a hybrid computer vision and ANN machine learning approach to enhance vortex core detection features and reduce false positives and negatives. The detection of vortices is one of the most critical and challenging characteristics of fluid flow analysis. In general, there are two traditional algorithms for detecting vortices: local and global detection which are based on velocity gradient tensor. Despite delivering fast results and being easy to implement, these techniques require careful selection of appropriate parameters in order to achieve reliable results, making it more difficult to identify vortex cores.

To deal with the issues in traditional vortex identification methods, we propose a hybrid computer vision and ANN machine learning approach. The initial step of identifying the vortex region takes place using YOLOv3 algorithm. This is followed by ANN to find the vortex core inside the vortex area. To examine the effectiveness of this model, we use the Taylor-Green vortex problem and Application to rotating wings-case which its flow field files obtain from [40]. The Taylor-Green vortex flow is simulated using Large-Eddy Simulation (LES) at Reynolds number 1600 [41], the large-scale turbulence is solved directly using Navier-Stokes equations, and the small-scale turbulence is modelled using the Smagorinsky-Lilly subgrid-scale model [42]. The discretisation of the Taylor-Green vortex was achieved using a uniform mesh having a resolution of 64^3 with a time-step of $\Delta t^* \approx 0.06$, and a total of 335 time steps are required to advance the solution from $t^* = 0$ to $t^* = 20$ at $CFL = 0.6$. In this case, t^* is a non-dimensional time, which is defined as $t^* = tU_0/L$. The initial conditions are as follows:

$$u = U_0 \sin \left(\frac{x}{L} \right) \cos \left(\frac{y}{L} \right) \cos \left(\frac{z}{L} \right) \quad (7)$$

$$v = -U_0 \cos \left(\frac{x}{L} \right) \sin \left(\frac{y}{L} \right) \cos \left(\frac{z}{L} \right) \quad (8)$$

$$w = 0, \quad (9)$$

$$p = p_0 + \frac{\rho_0 U_0^2}{10} \left(\cos \left(\frac{2x}{L} \right) + \cos \left(\frac{2y}{L} \right) \right) \left(\cos \left(\frac{2z}{L} \right) + 2 \right) \quad (10)$$

In this case, taking the kinematic viscosity to $\nu = 1/1600 \text{ m}^2 \text{ s}^{-1}$ in order to achieve the desired Reynolds number of 1600. To do this, set $U_0 = 1 \text{ m s}^{-1}$, $\rho_0 = 1 \text{ kg m}^{-3}$, and $p_0 = 0 \text{ Pa}$. Fig 5 shows the contour velocity plot at the beginning ($t^* = 0$) at the left-hand side and the final time step ($t^* = 20$) at the right-hand side.

Validation is undertaken by comparing our simulation set-up with

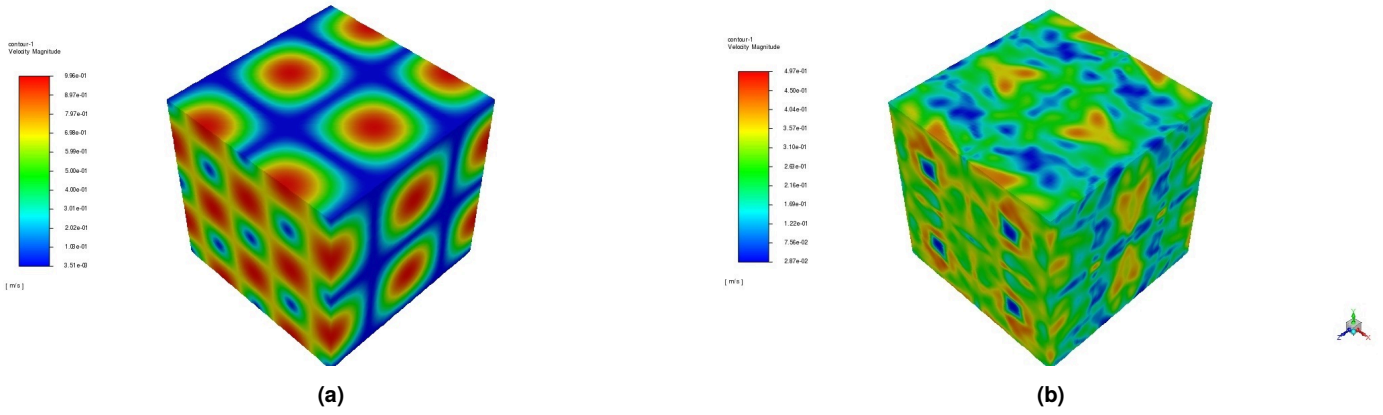


FIGURE 5: CONTOUR VELOCITY PLOT. THE LEFT-HAND SIDE (A) AT THE BEGINNING INITIAL STATE AND THE RIGHT-HAND SIDE (B) AT THE END TIME STEP.

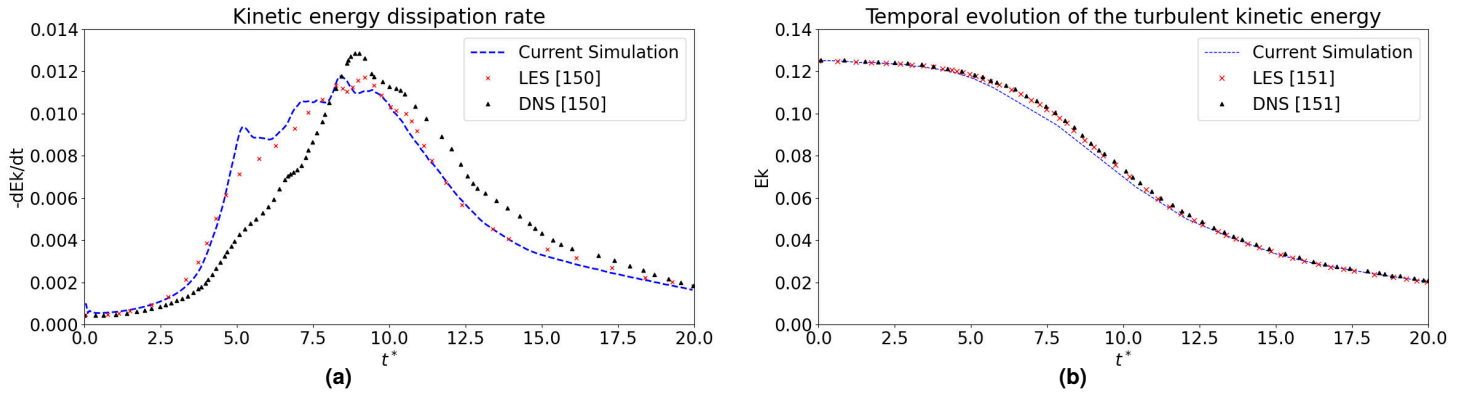


FIGURE 6: ON THE LEFT-HAND SIDE (A) IS THE KINETIC ENERGY DISSIPATION RATE AND TEMPORAL EVOLUTION OF THE TURBULENT KINETIC ENERGY IS SHOWN ON THE RIGHT-HAND SIDE (B), WHERE THE REFERENCE LARGE EDDY SIMULATION (LES) AND DIRECT NUMERICAL SIMULATION (DNS) WAS TAKEN FROM [43] AND FROM [44] RESPECTIVELY, AND THE CURRENT SIMULATION RESULTS SHOWN IN THE DASHED BLUE LINE WERE OBTAINED USING LES ON A 64^3 GRID AT A REYNOLDS NUMBER OF $Re=1600$.

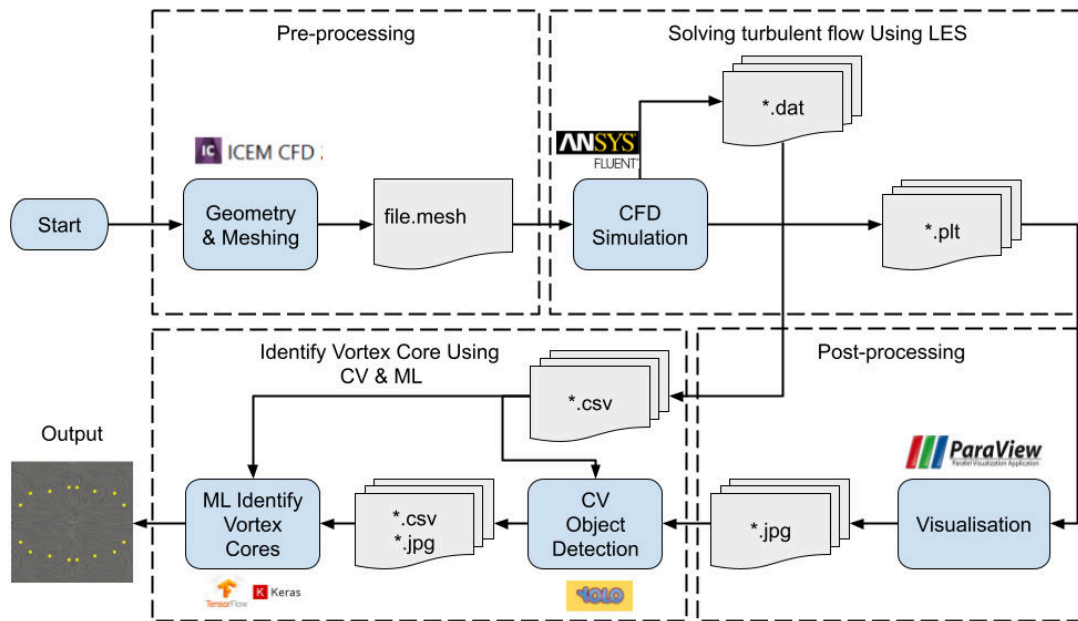
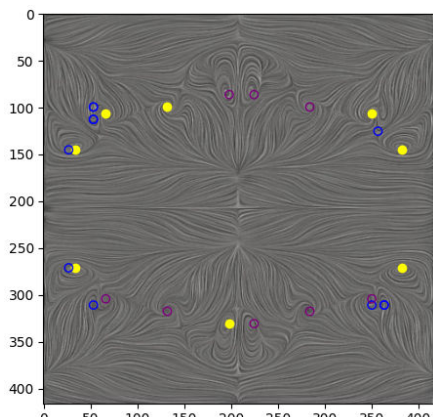
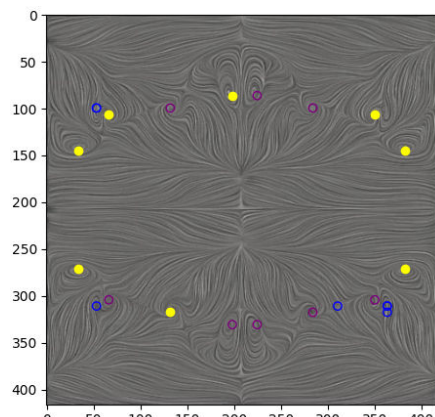


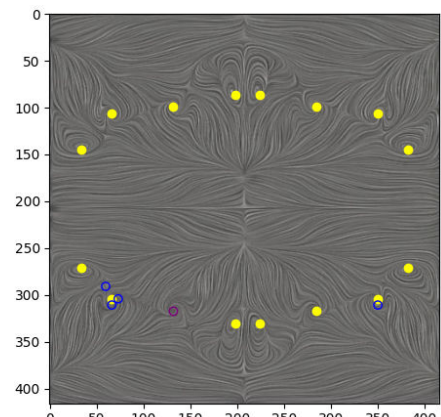
FIGURE 7: PROCESS FLOW DIAGRAM ARCHITECTURE TO IDENTIFY VORTEX CORES.



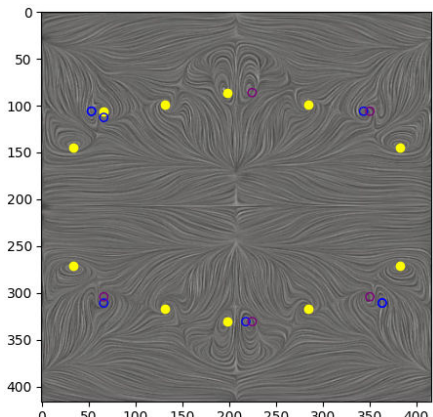
Variable set 1 Number of flow field files 5



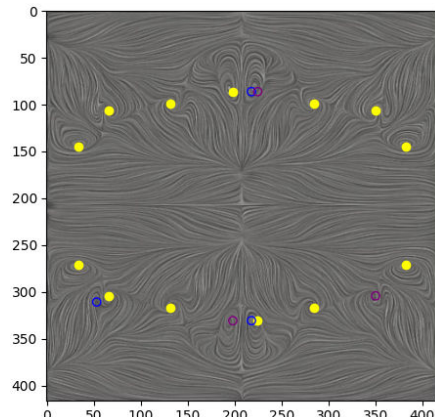
Variable set 1 Number of flow field files 10



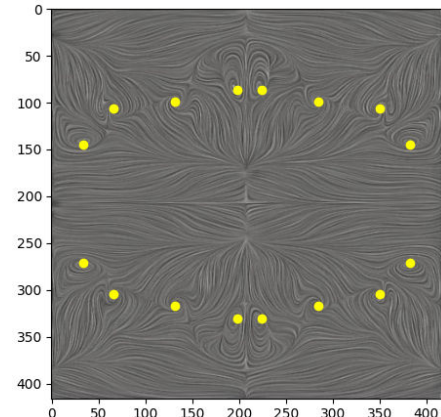
Variable set 1 Number of flow field files 25



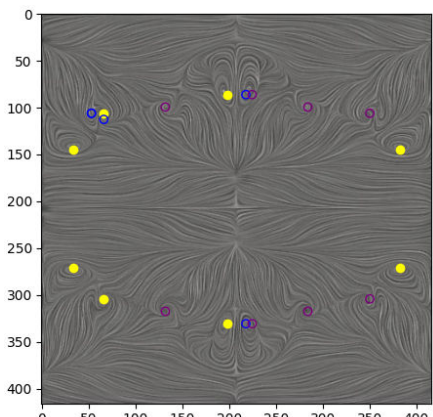
Variable set 2 Number of flow field files 5



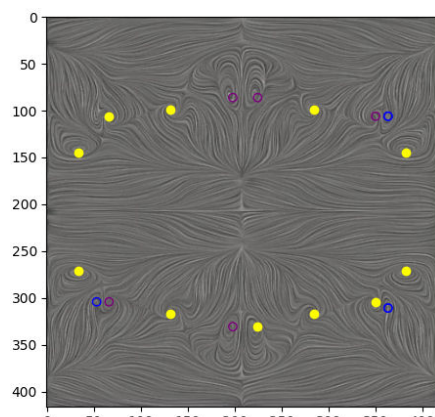
Variable set 2 Number of flow field files 10



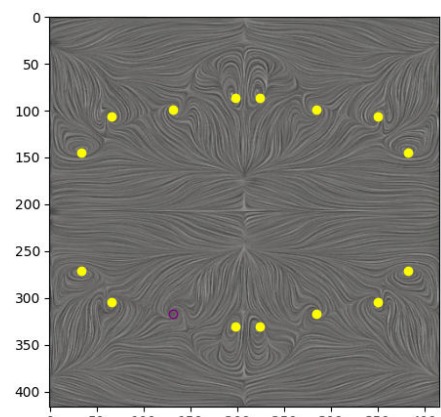
Variable set 2 Number of flow field files 25



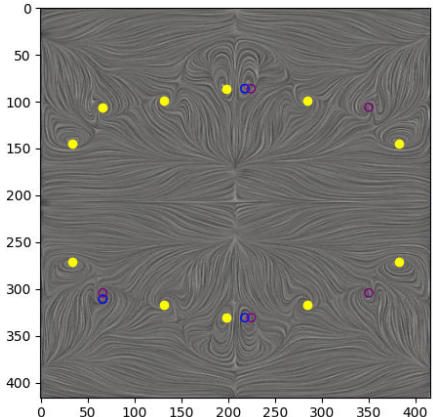
Variable set 3 Number of flow field files 5



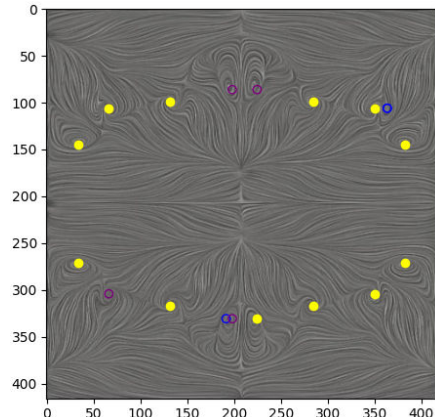
Variable set 3 Number of flow field files 10



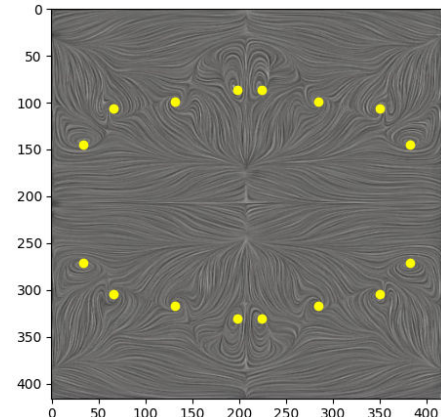
Variable set 3 Number of flow field files 25



Variable set 4 Number of flow field files 5



Variable set 4 Number of flow field files 10



Variable set 4 Number of flow field files 25

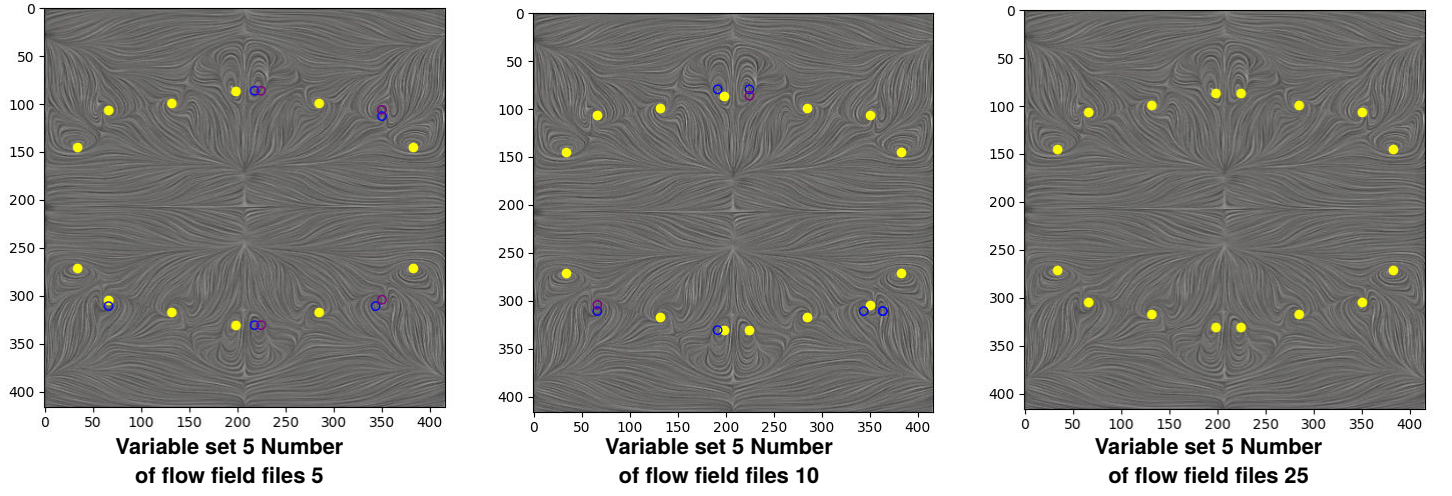
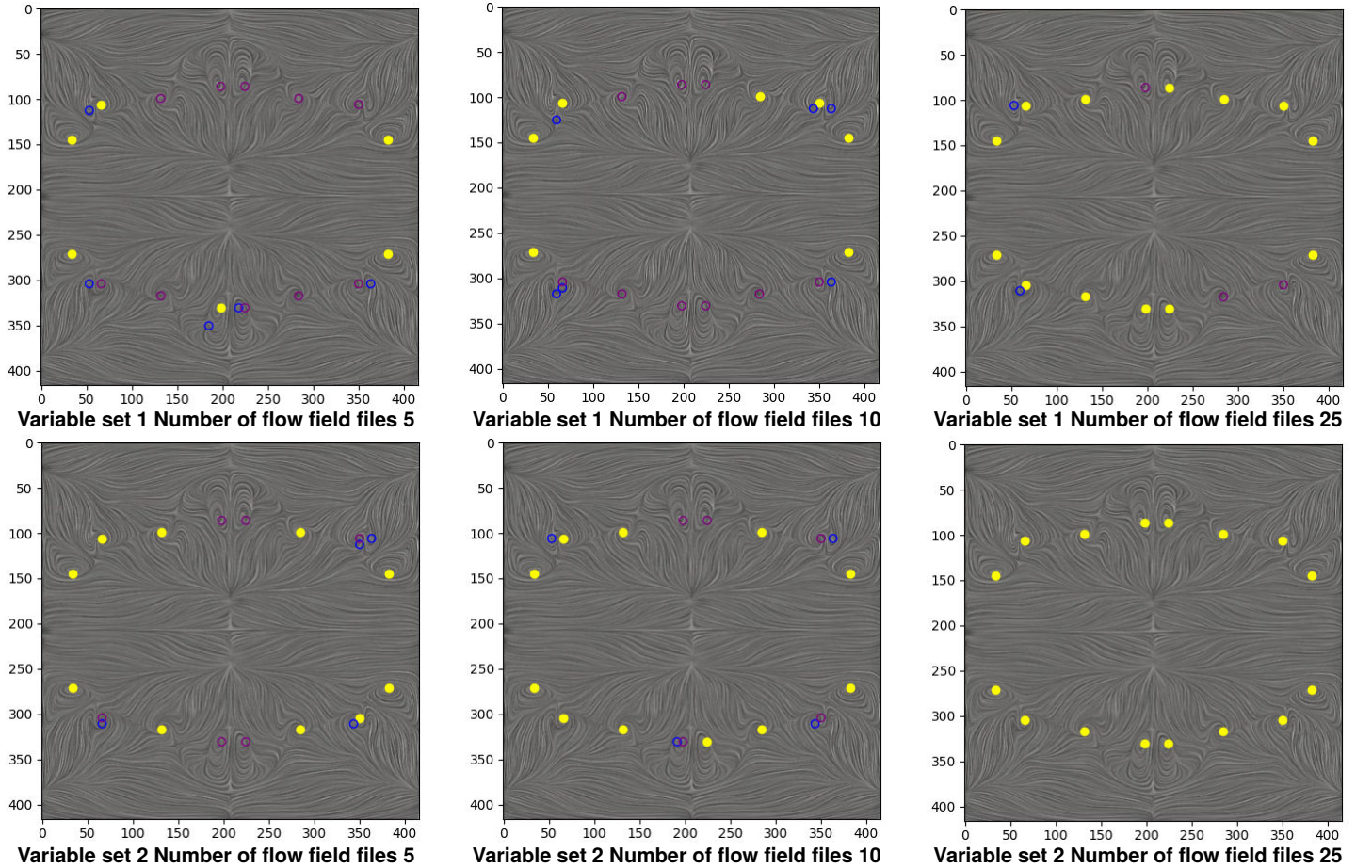


FIGURE 8: THE FIGURES SHOW THE RESULT OF USING THE HYBRID APPROACH (YOLO, ANN) FOR DIFFERENT FLOW FIELD FILES AND DIFFERENT VARIABLE SETS WHERE THE YELLOW DOTS SHOW THE RIGHT PREDICTION OF ACTUAL VORTEX CORES (TRUE POSITIVES) AND BLUE DOTS ARE FALSE POSITIVES AND THE PURPLE DOTS INDICATE THE FALSE NEGATIVES.



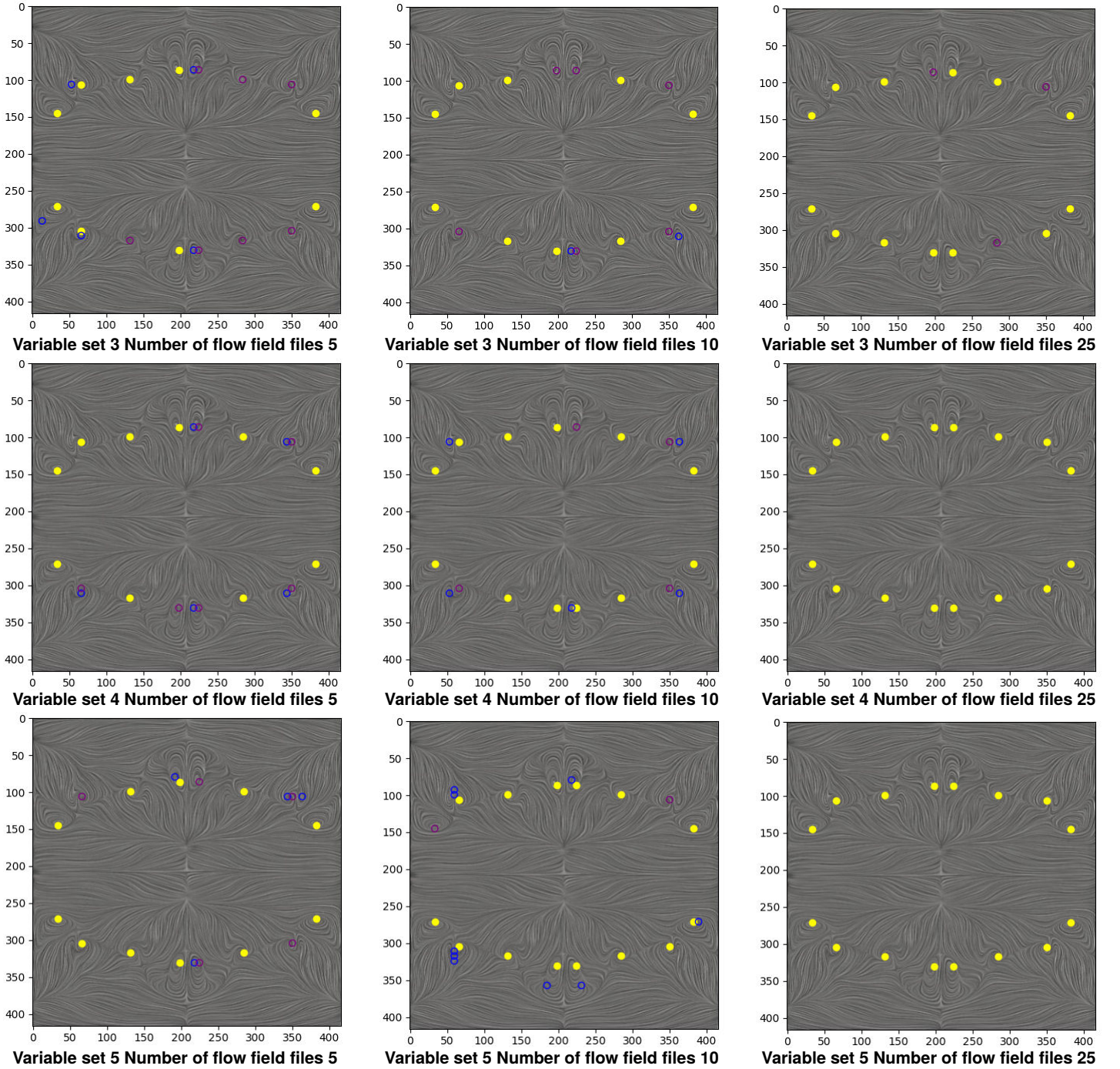


FIGURE 9: THE FIGURES SHOW THE RESULT OF USING THE SINGLE APPROACH (PURE ANN) FOR DIFFERENT FLOW FIELD FILES AND DIFFERENT VARIABLE SETS WHERE THE YELLOW DOTS SHOW THE RIGHT PREDICTION OF ACTUAL VORTEX CORES (TRUE POSITIVES) AND BLUE DOTS ARE FALSE POSITIVES AND THE PURPLE DOTS INDICATE THE FALSE NEGATIVES.

DeBonis [44] and Beck and Gassner [43] studies based on kinetic energy and time evolution of kinetic energy. In order to compute the temporal evolution of kinetic energy, the instantaneous velocity vector is taken into account as follows:

$$E_k = \frac{1}{\rho_0 V} \int_V \rho \frac{\mathbf{U} \cdot \mathbf{U}}{2} dV \quad (11)$$

The temporal evolution of the kinetic energy dissipation rate is

computed as:

$$\varepsilon = -\frac{dE_k}{dt} \quad (12)$$

The kinetic energy dissipation rate and the temporal evolution of the turbulent kinetic energy are shown in Fig 6 on the left and right-hand sides respectively. Based on the same mesh size as our simulation, we compare our data here against LES results and direct numerical simulation (DNS) results. The architec-

TABLE 1: SHOWING A NUMBER OF EXPECTED AND DETECTED VORTICES FOR THE SAMPLE TEST, AS WELL AS COMPARISON OF HYBRID APPROACHES AND A SINGLE APPROACH HOW TRUE POSITIVE (TP) FALSE POSITIVE (FP) AND FALSE NEGATIVE (FN) REDUCE AS THE NUMBER OF FLOW FIELD FILES IN THE TRAINING DATA SET INCREASES WITH DIFFERENT VARIABLE SETS FROM 1 TO 5 AS INPUT FEATURES MENTION IN SECTION 3.4 AND THE COMPUTATIONAL TIME IS TAKEN FOR TRAINING IN BOTH METHODS.

# of files	Variable set	Total vortices	Test image							
			Both approaches			Training time	Single approach			Training time
			TP	FP	FN		TP	FP	FN	
5	1	16	8	11	8	00:42:27	6	5	10	01:39:33
	2	16	11	8	5	00:37:48	10	4	6	01:43:24
	3	16	8	5	8	00:38:28	9	5	7	01:46:18
	4	16	11	3	5	00:42:10	10	5	6	01:53:45
	5	16	12	5	4	00:43:18	11	4	5	01:59:37
10	1	16	8	7	8	01:41:59	7	6	9	02:57:52
	2	16	13	3	3	01:44:38	11	4	5	03:12:14
	3	16	11	5	5	01:46:02	10	2	6	03:20:09
	4	16	12	3	4	01:51:11	12	5	4	03:24:20
	5	16	14	7	2	01:55:28	14	9	2	03:36:43
20	1	16	13	9	3	02:24:50	10	8	6	05:38:34
	2	16	15	2	1	02:40:48	13	4	3	05:42:27
	3	16	14	1	2	02:33:17	12	6	4	05:40:01
	4	16	15	3	1	02:47:10	13	2	3	05:49:52
	5	16	16	0	4	02:56:19	15	3	1	05:54:11
25	1	16	15	4	1	03:11:16.	13	2	3	07:53:40
	2	16	16	0	0	03:23:38	16	0	0	08:00:03
	3	16	15	0	1	03:15:19	13	0	3	07:58:06
	4	16	16	0	0	03:48:58	16	0	0	08:13:13
	5	16	16	0	0	04:04:18	16	0	0	08:34:01

ture diagram in Fig 7 illustrates the end-to-end process through which different software and tools are used to identify vortices for various purposes. A geometry of Taylor-Green vortex and mesh refinement is created using ICEM CFD as part of the Pre-processing. With the mesh file generated, the LES model is used to solve the turbulent flow using Ansys Fluent 2019 software to produce Tecplot files for visualisation and flow field files with numerical data such as velocity, pressure, etc to be used in the ANN machine learning algorithm. Then we generate streamline plots for x, y, and z planes in our computational domain using ParaView which we extract the velocity vector and then compute local streamlines for each pixel with the LIC algorithm, as input to our computer vision framework. Once the images have been labelled, we pass them to the YOLOv3 algorithm of the computer vision framework to identify the region of vortices. After detecting the areas of vortical structures using computer vision, we apply ANN machine learning to locate the vortex core within each vortex region. Within these regions which were identified by YOLO, we obtain the values of different sets of features to train our model using ANNs (TensorFlow 2, Keras). We begin with only velocity and pressure variables, then add more derived quan-

ties (velocity gradients, pressure gradients, Q-criterion, vorticity, and magnitudes of vector quantities), this study is composed of five different sets of variables is shown in the previous sub section 3.4. Due to this, a hybrid approach decreases the training time for all tested cases, and also it reduces the number of false positives and negatives where we can see the comparison and difference between the hybrid approach and the single approach (pure ANN machine learning) in the result section.

5. OUR NUMERICAL RESULTS

5.1 Taylor-Green vortex

For testing our model, we choose a flow field file exported from the Taylor-Green vortex case among the 335-time steps in the symmetry plane in the x, y, and z-direction, which is not part of the training data set. The results show the predicted vortices using our hybrid approaches (YOLOv3 and ANN algorithm) and single approach (pure ANN machine learning) with different input variable combinations for the ANN machine learning part to detect vortex cores.

Fig 8 and Fig 9 illustrate the results based on using the hybrid approach and the single approach (pure ANN) for a variety

of variable sets and different flow field files respectively. The yellow dots depict where a vortex core has been correctly detected (true positives), and the blue dots indicate where a vortex core has been wrongly identified (false positives). The purple dots represent when a vortex core has not been identified (false negatives). These figures reveal where true positives, false positives, and false negatives are located. It is also noticeable that the false positives and negatives in this test case are located in the same region of the vortex for the hybrid approach and pure machine learning.

Table 1 shows the number of flow field files, variable sets, true positives, false positives, and negatives for hybrid and single approaches, as well as the training time for each. Table 1 shows the result of using 5, 10, 20, and 25 different flow field files for the ANN machine learning step for both the hybrid and single approaches for each variable set. We see that using just five flow field files as a data set, the false positive and negative rates remain high for both approaches and just using a single (pure ANN) gives fewer false positives. At 20 flow field files with variable set 5, our hybrid approach produces zero false positives, and at this point, with half the training time, the hybrid approach is starting to produce fewer false positives than the single approach. Once the training data set has 25 flow field files, neither approach produces false positives or negatives for 2, 4, and 5 variable sets. Our findings show that the hybrid approach of using YOLO for identifying the regions of the vortex core and ANN for detecting the vortex core within these regions is more efficient and faster than using pure ANN to train our models. In addition, we can conclude that using gradients or variable set 2 always produces better results than applying variable set 1. Traditional methods use only velocity gradient tensors, whereas we also utilised absolute velocity and pressure components.

Fig 10 demonstrates the true positives for different Variable sets relative to flow field files using the hybrid approach with solid line and pure ANN with dash line. With each variable set represented by a different colour. As we can see in Fig 10 variable set one has the strongest gradient, so it detects true positives the quickest after 10 images. In contrast, variable set two has a weaker gradient but more true positives initially. Despite variable set two not increasing as strongly as variable set one, it has the best compromise between the number of input images and accuracy and detects more true positives right away as a result. The rest of the variable sets lie in between, yet in the end, they provide the 16 true positives we are looking for. In general, for both approaches, it is evident that the true positive curve is increasing in proportion to an increasing number of flow field files for all variable sets. Nevertheless, the hybrid approach performs slightly better than pure ANN machine learning in capturing all vortex cores, it requires 20 flow field files for variable set four, whereas pure ANN requires 25 flow field files for variable sets two, four, and five. False positives and negatives with flow field files are illustrated in Fig 11 and Fig 12 respectively, for various variable sets as determined by the hybrid approach with solid line and single approach with dash line. According to both Fig 11 and Fig 12, variable set one has the highest false positives and negatives, and this is why Fig 10 shows that true positives dramatically increase with variable set one while variable set two is always one of the

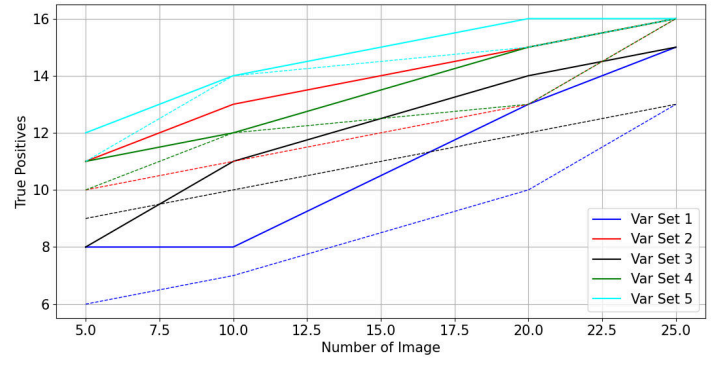


FIGURE 10: THE VALUES OF TRUE POSITIVE FOR DIFFERENT VARIABLE SETS WITH RESPECT TO NUMBER OF FLOW FIELD FILES BASED ON HYBRID APPROACHES WITH SOLID LINE AND SINGLE APPROACH (PURE ANN) WITH DASH LINE.

lowest. Therefore, variable set 1 is indifferent to the hybrid and pure machine learning approaches, whereas all other variables are benefited from computer vision.

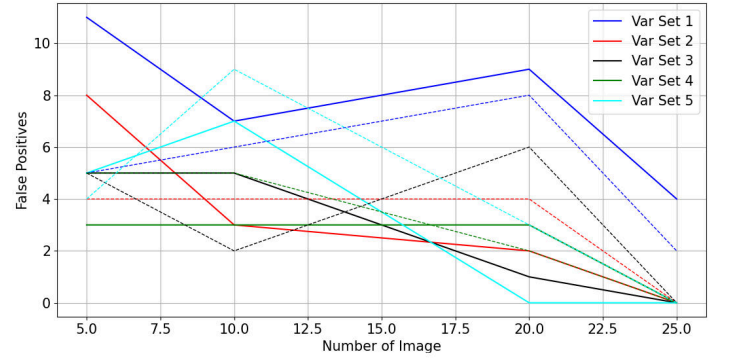


FIGURE 11: THE VALUES OF FALSE POSITIVE FOR DIFFERENT VARIABLE SETS WITH RESPECT TO THE NUMBER OF FLOW FIELD FILES BASED ON HYBRID APPROACHES WITH A SOLID LINE AND SINGLE APPROACH (PURE ANN) WITH DASH LINE.

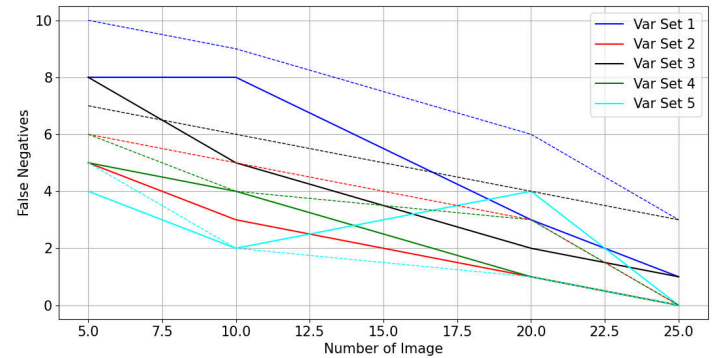


FIGURE 12: THE VALUES OF FALSE NEGATIVES FOR DIFFERENT VARIABLE SETS WITH RESPECT TO THE NUMBER OF FLOW FIELD FILES BASED ON HYBRID APPROACHES WITH SOLID LINE AND SINGLE APPROACH (PURE ANN) WITH DASH LINE.

5.2 The Impact of Noise Magnitude on Detection Model Performance

The robustness of our model against noisy input is a critical factor to consider in evaluating its real-world applicability. To assess this, an examination of the model's dependence on the noise magnitude is essential. Noise, in this context, refers to any unwanted or random variations in the input data that can potentially affect the model's performance. A robust model should ideally exhibit consistent and reliable detection performance even when faced with noisy input, as demonstrated by the introduction of Gaussian noise in the illustrated Figure 13. Figure 14 and Fig-

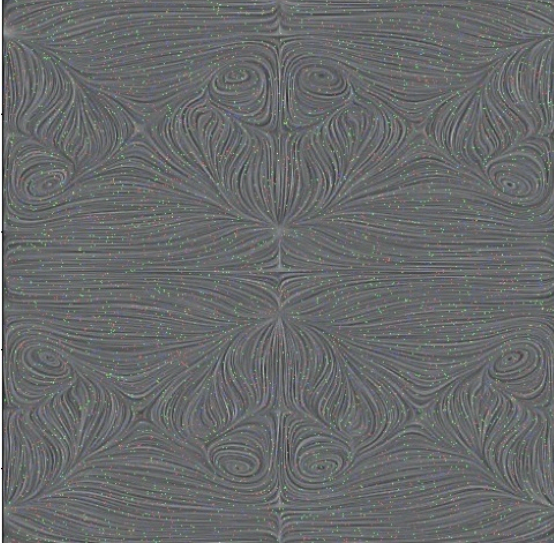


FIGURE 13: VISUALISATION OF TEST IMAGE WITH NOISE MAGNITUDE 1: NOISY INPUT REPRESENTATION.

ure 15 present the results of computer vision and machine learning respectively. Figure 14 demonstrates the outcomes of computer vision, showcasing the model's remarkable ability to identify vortex regions of the test image with a noise magnitude value of one. Simultaneously, Figure 15 highlights the machine learning validation to detect all vortex cores within their regions. Remarkably, both figures underscore the model's proficiency in accurately detecting vortex cores, even when faced with noise in the input test image at a magnitude value of one.

The table 2 presents the performance metrics of a detection algorithm under different noise magnitudes. The key metrics include True Positive, False Positive, False Negative, Precision, Recall, and F1 Score, with each row corresponding to a specific noise magnitude. The precision rate measures the accuracy of the predictions, and the calculation formula is as follows:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \times 100 \quad (13)$$

Meanwhile, recall measures how well all the true positives are found, which are defined as:

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \times 100 \quad (14)$$

The F1 score provides a single metric that balances both precision

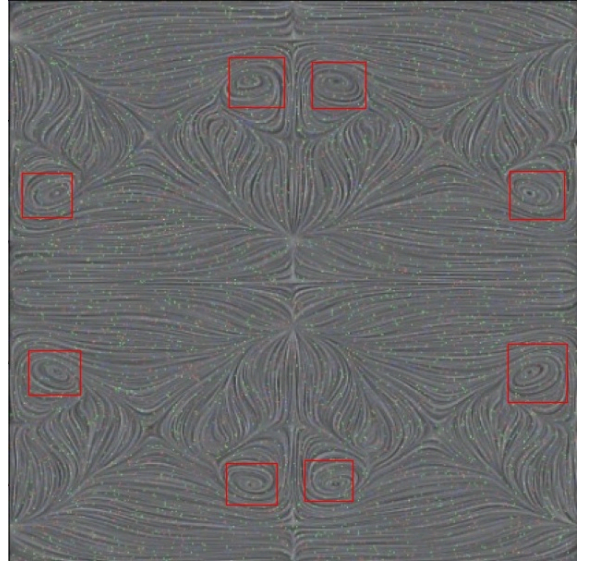


FIGURE 14: COMPUTER VISION ANALYSIS OF VORTEX CORE DETECTION IN NOISY INPUT TEST IMAGE

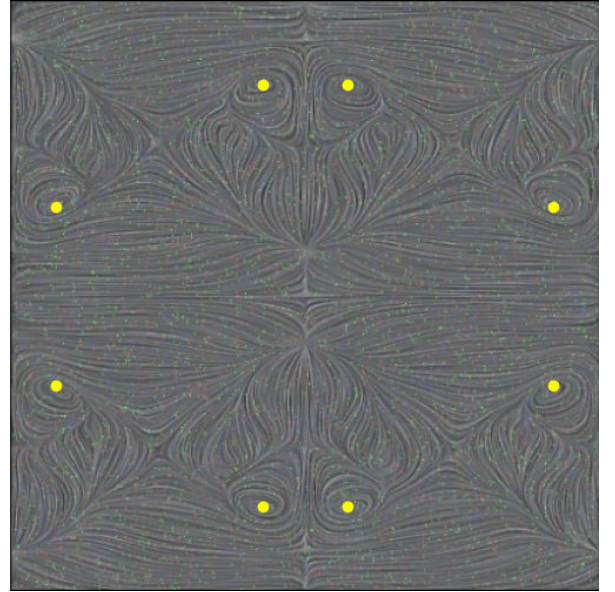


FIGURE 15: MACHINE LEARNING VALIDATION OF VORTEX CORE IDENTIFICATION IN NOISY INPUT TEST IMAGE

and recall, which are defined as:

$$\text{F1 Score} = 2 \times \left(\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \right) \quad (15)$$

For Noise Magnitudes 0.5 and 1.0, the algorithm demonstrates perfect performance with 100% Precision, Recall, and F1 Score. This indicates accurate detection without any false positives or false negatives. However, for Noise Magnitude 1.2, while Precision remains high, Recall decreases, resulting in a lower F1 Score. This suggests some missed positive instances. Noise Magnitude 1.5 shows a decrease in all metrics, indicating a more challenging detection scenario with increased false positives and false negatives. Notably, for Noise Magnitude 2.0, there are no

True Positives or False Negatives, resulting in undefined values for Precision, Recall, and F1 Score. This might imply that the algorithm failed to detect any positives under this noise magnitude. Overall, the table provides a comprehensive overview of the detection algorithm's performance under varying noise magnitudes, offering valuable insights for further analysis and improvement of the algorithm's robustness.

5.3 The Taylor-Green vortex using automatic labelling

One of the limitations of supervised deep learning is that large, manually labelled data sets are required. Manual labelling of geometries with finer meshes is inefficient because not all samples are equal, and it is hard to determine where vortex cores are located. Therefore, we utilised computer vision and the K-means clustering algorithm to automatically label the data set. In Fig 16, the left side shows automatically identifying regions close to vortex cores is done using computer vision. The computer vision results are manifested in the different-sized boxes surrounding vortex cores (red square). These regions are then divided into k groups using the K-means clustering algorithm. We use the silhouette score library to determine the number of clusters K and assume that each cluster has a centroid at its centre. The right-hand side of Fig 16 shows the result after using computer vision and the K-mean clustering algorithm for automatic labelling. Blue stars represent clustering coordinates, and yellow dots represent cluster centres. The cluster centre's coordinates are defined as vortex cores (one) in the label field on the data set of flow field files, otherwise, they are set to zero. By labelling the data set automatically, we are able to prepare the data set efficiently and quickly for training our model to identify vortex cores. Based on table 3, the Mean Square Error (MSE) of vortex core coordinates using manual and automatic labelling techniques for the same test image with eight vortex cores (not included in the data set) is 0.45%. The rather low MSE value suggests that our automatic labelling approach is suitable for labelling large data sets. We further test this hypothesis in the next section, where we apply our new automatic labelling to a steady-state, rotating wing test case.

5.4 Application to rotating wings-case

Our model is tested using the real-world data from Hovering Rotor Solutions (Caradonna and Tung) [45]. Caradonna and Tung conducted their experiment at NASA Ames Research Centre in a controlled wind tunnel using a two-bladed rotor. In this study, we use published data of Ricci *et al.* [46], who have provided numerical simulations for this test case. The domain is cylindrical-shaped, with a radius of 40m and length of 80m, and the helicopter blade is located in the middle as shown in Fig 17. The blade is derived from an untwisted and untapered NACA 0012 aerofoil with an aspect ratio of 6 and a radius of 1.143m. The simulations were conducted at a blade tip Mach number of $M_{tip} = 0.89$, a pitch angle of $\theta_c = 8^\circ$, and an angular velocity of 265.9 rad/s. The published CFD data from [46] is used as training data, and we use ParaView to extract 10 slices with different angles, then clip all images near the rotor to obtain the flow field files for training the model. Then we label the data set automatically by using computer vision which identifies the

regions close to the vortex core and uses our K-Means clustering algorithm to find clusters, from which we then obtain the centre of clusters which are labelled as vortex cores.

Hybrid and single approaches are compared in Table 4, including the number of flow field files in the data set, feature variable sets, true positives, false positives, and negatives and their training times. As shown in table 4, the results of pure ANN and hybrid approaches using 1, 5, and 10 different flow field files were performed for each variable set in the ANN machine learning step. According to the table 4, variable set two is an acceptable alternative and achieves almost as good accuracy as the results of the first test case in table 1. However, variable set two would require slightly more flow field files in the data set to converge the training and get down the false positives and false negatives. As a result, it is possible to conclude that variable set 2, which consists of velocity, pressure, and their derivative, always produces more accurate results than variable set 1. Similarly, table 4 shows that the hybrid approach achieves zero false positives and negatives for 10 flow field files with variable sets 4 and 5, whereas the pure ANN algorithm produces zero false positives and negatives for 10 flow field files with variable set 5. However, hybrid method training requires less time than pure ANN machine learning.

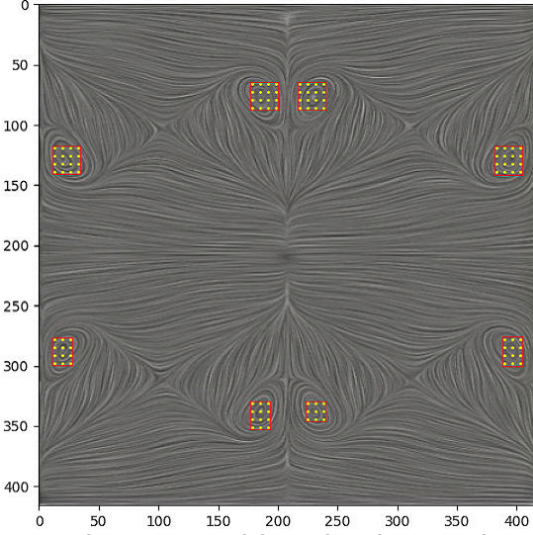
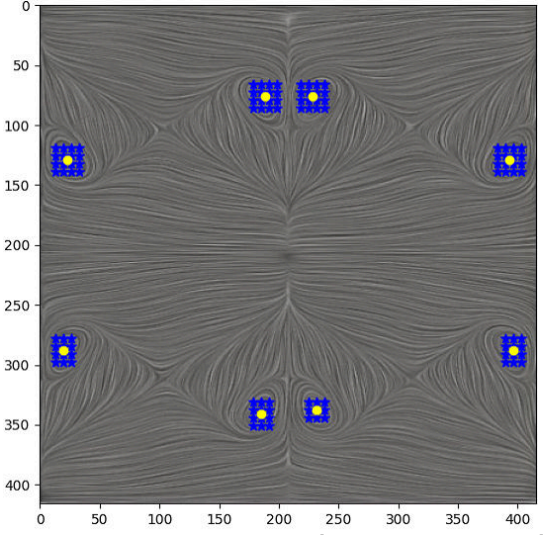
Fig 18 and Fig 19 depict the results obtained using the pure ANN and hybrid approach for variable set five and different flow field files respectively. As can be seen in the figures, the false positives obtained with pure ANN are far from vortex cores, whereas the false positives obtained with the hybrid approach are close to vortex cores. Thus, this is another advantage of utilising a hybrid approach, other than reducing training time.

6. CONCLUSIONS

Detection of vortex cores within CFD is still a challenging task. In this project, we introduce a hybrid computer vision (YOLOv3 algorithm) and ANN machine learning (TensorFlow, Keras) approach to detect vortex cores for the Taylor-Green vortex and rotating wing cases. The data set of Taylor Green Vortex is prepared using both manual and newly proposed automatic labelling methods. For automatic labelling methods, we utilised computer vision and the k-means clustering algorithm to locate the centre of the cluster as the vortex core. The result shows that the MSE between manual and automatic labelling techniques is low and thus a viable alternative to labour-intensive manual labelling. We have applied this new automatic labelling approach to a rotating wing case which showed similar trends compared to the Taylor-Green vortex problem. The first step is to employ computer vision to identify the region where the vortices are in order to present only vortical flow patterns to the machine learning step, which is executed after regions of vortices are identified to extract the location of the vortex core. In this study, we have shown that velocity and pressure are essential information to predict the vortex cores correctly and that we can reduce the rate at which the false positives and negatives are reduced by incorporating gradient information of the velocity and pressure. In addition, the hybrid approach was able to speed up detection by up to 3 times compared to the pure ANN machine learning approach, particularly for the application to the

TABLE 2: PERFORMANCE METRICS OF DETECTION ALGORITHM UNDER DIFFERENT NOISE MAGNITUDES.

S	Noise Magnitude	True Positive	False Positive	False Negatives	Precision (%)	Recall (%)	F1 Score (%)
1	0.5	8	0	0	100.00	100.00	100.00
2	1.0	8	0	0	100.00	100.00	100.00
3	1.2	5	1	3	83.33	62.50	71.43
4	1.5	3	4	5	42.86	37.50	40.00
5	2.0	0	0	0	∞	∞	∞

**The result of using computer vision to identify the regions close to vortex cores.****Blue stars are shown as the culture points, and yellow dots indicate cluster centres.****FIGURE 16: THE FIGURES SHOW THE RESULT OF USING THE COMPUTER VISION AND THE K-MEANS CLUSTERING ALGORITHM TO IDENTIFY VORTEX CORES FOR LABELLING DATA SETS AUTOMATICALLY.****TABLE 3: VORTEX CORE COORDINATES USING MANUAL AND AUTOMATIC LABELLING TECHNIQUES FOR THE SAME IMAGE TEST.**

Vortex core coordinates									
manual	x	0.305	2.815	3.469	5.978	0.305	2.815	3.469	5.978
	y	4.339	5.101	5.101	4.339	1.944	1.182	1.182	1.944
Auto	x	0.299	2.790	3.540	5.980	0.299	2.790	3.490	5.978
	y	4.340	5.190	5.190	4.290	1.990	1.100	1.200	1.990

rotating wing case, and to reduce false positives and negatives more quickly than using just the single approach.

Therefore, we can conclude that our newly developed hybrid computer vision + ANN approach is not only able to improve the detection time significantly but also reduce the pre-processing time through our new K-means clustering-based automatic labelling approach.

Additional investigations with the framework for cases in which training data is limited needs to be further explored as a future step to understand the robustness of the framework for

cases with limited training data.

Acknowledgments

The author Hazem Ashor Amran Abolholl has received funding through the Academic Relations Department of the Libyan Embassy – London, and their financial support is greatly acknowledged.

REFERENCES

- [1] Hunt, J. C. R., Wray, A. A. and Moin, P., 1988. "Eddies, Streams, and Convergence Zones in Turbulent Flows." *Center for Turbulence Research, Proceedings of the Sum-*

TABLE 4: SHOWING A NUMBER OF EXPECTED AND DETECTED VORTICES FOR THE APPLICATION TO ROTATING WINGS TEST CASE, AS WELL AS COMPARISON OF HYBRID APPROACHES AND A SINGLE APPROACH HOW FALSE POSITIVES AND NEGATIVES REDUCE AS THE NUMBER OF FLOW FIELD FILES IN THE TRAINING DATA SET INCREASES WITH DIFFERENT VARIABLE SETS FROM 1 TO 5 AS INPUT FEATURES MENTION IN SECTION 3.4 AND THE COMPUTATIONAL TIME IS TAKEN FOR TRAINING IN BOTH METHODS.

# of files	Variable set	Total vortices	Test image							
			Both approaches			Training time	Single approach			Training time
			TP	FP	FN		TP	FP	FN	
1	1	4	0	4	4	00:28:32	0	5	4	01:19:43
	2	4	1	5	3	00:31:16	0	4	4	01:13:53
	3	4	0	3	4	00:34:13	0	5	4	01:23:12
	4	4	1	2	3	00:37:21	0	3	4	01:26:42
	5	4	2	2	2	00:35:29	1	4	3	01:39:12
5	1	4	1	2	3	00:51:02	1	3	3	01:53:16
	2	4	2	3	2	00:55:17	2	4	2	02:24:38
	3	4	1	3	3	00:57:29	1	4	3	02:19:58
	4	4	3	3	1	00:53:27	2	2	2	02:16:52
	5	4	4	2	0	00:56:43	3	3	1	02:53:36
10	1	4	2	3	2	01:48:50	1	2	3	04:53:11
	2	4	3	2	1	01:55:14	2	3	2	05:04:35
	3	4	2	2	2	01:50:37	2	3	2	05:08:11
	4	4	4	0	0	01:53:20	4	1	0	05:12:12
	5	4	4	0	0	01:59:54	4	0	0	05:14:11



FIGURE 17: FIGURE DEPICTS THE QUARTER DOMAIN OF THE APPLICATION TO ROTATING WINGS (CARADONNA AND TUNG ROTOR) CASE.

- [2] Chong, M. S., Perry, A. E. and Cantwell, B. J., 1990. "A General Classification of Three-Dimensional Flow Fields." *Phys. Fluids A*, 2(5), pp. 765-777 .
- [3] Liu, C. Q., Wang, Y. Q., Yang, Y. and Duan, Z. W., 2016. "New Omega Vortex Identification Method." *Sci. China: Phys., Mech. Astron.*, 59(8), pp. 1-19 .
- [4] Zhou, J., Adrian, R. J., Balachandar, S. and Kendall, T. M., 1999. "Mechanisms for Generating Coherent Packets of Hairpin Vortices in Channel Flow." *J. Fluid Mech.*, 387, pp. 353-396 .
- [5] Abolholl, Hazem Ashor Amran, Teschner, Tom-Robin and Moulitsas, Irene. , 2023. "SURFACE LINE INTEGRAL CONVOLUTION-BASED VORTEX DETECTION USING COMPUTER VISION." *Journal of Computing and Information Science in Engineering*, pp. 1-18 .
- [6] Redmon, J., Divvala, S., Girshick, R. and Farhadi, A., 2016. "You Only Look Once: Unified, Real-Time Object Detection." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Las Vegas, NV, June 27-30, pp. 779-788.
- [7] Abadi, Martín, Barham, Paul, Chen, Jianmin, Chen, Zhifeng, Davis, Andy, Dean, Jeffrey, Devin, Matthieu, Ghemawat, Sanjay, Irving, Geoffrey, Isard, Michael, Kudlur, Manjunath, Levenberg, Josh, Monga, Rajat, Moore, Sherry, Murray, Derek G., Steiner, Benoit, Tucker, Paul, Vasudevan,

mer Program. No. 1970, Stanford University, Dec 1, pp. 193-208.

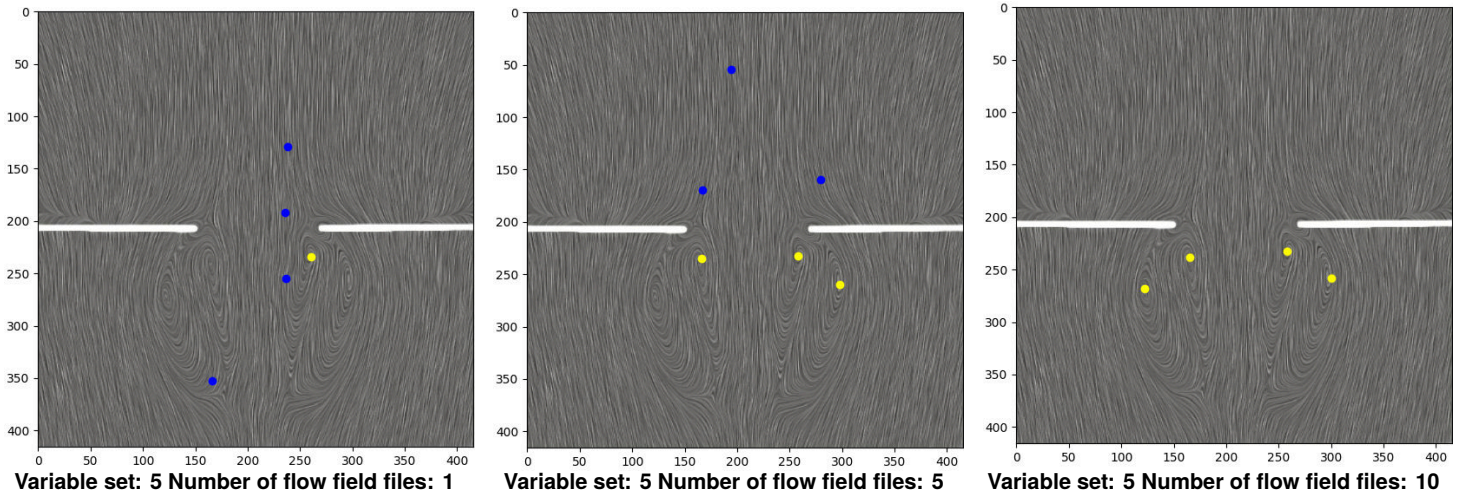


FIGURE 18: THE FIGURES SHOW THE RESULT OF USING THE SINGLE APPROACH (PURE ANN) FOR DIFFERENT FLOW FIELD FILES AND FIVE VARIABLE SETS WHERE THE YELLOW DOTS SHOW THE RIGHT PREDICTION OF ACTUAL VORTEX CORES (TRUE POSITIVES) AND THE BLUE DOTS ARE FALSE POSITIVES.

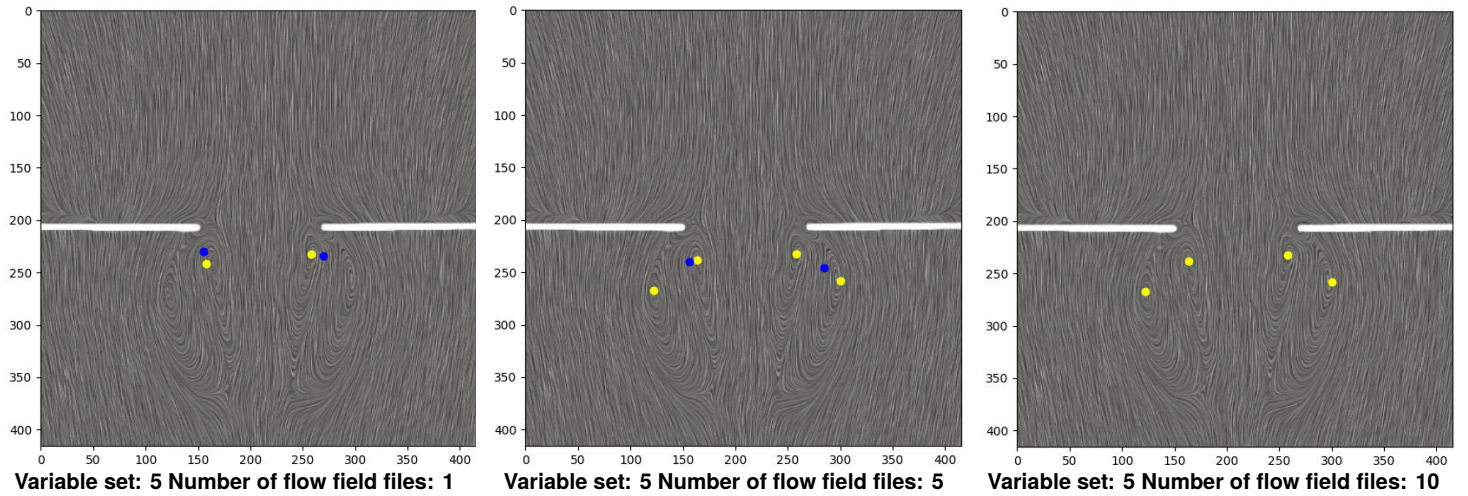


FIGURE 19: THE FIGURES SHOW THE RESULT OF USING THE HYBRID APPROACH (YOLO, ANN) FOR DIFFERENT FLOW FIELD FILES AND FIVE VARIABLE SETS WHERE THE YELLOW DOTS SHOW THE RIGHT PREDICTION OF ACTUAL VORTEX CORES (TRUE POSITIVES) AND BLUE DOTS ARE FALSE POSITIVES.

- Vijay, Warden, Pete, Wicke, Martin, Yu, Yuan and Zheng, Xiaoqiang, 2016. "TensorFlow: A system for large-scale machine learning." *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2016*, pp. 265-283.
- [8] Kohler, B., Gasteiger, R., Preim, U., Theisel, H., Gutberlet, M. and Preim, B., 2013. "Semi-Automatic Vortex Extraction in 4D PC-MRI Cardiac Blood Flow Data Using Line Predicates." *IEEE Trans. Vis. Comput. Graph.*, **19**(12), pp. 2773-2782 .
- [9] Oeltze-Jafra, S., Cebal, J. R., Janiga, G. and Preim, B., 2016. "Cluster Analysis of Vortical Flow in Simulations of Cerebral Aneurysm Hemodynamics." *IEEE Trans. Vis. Comput. Graph.*, **22**(1), pp. 757-766 .
- [10] Kenwright, D. and Haines, R., 1997. "Vortex Identification—Applications in Aerodynamics: A Case Study." *Proceedings of the IEEE Visualization Conference*. Phoenix, AZ, Oct. 19-24, pp. 413-415.
- [11] Gutak, A. D., 2015. "Experimental Investigation and Industrial Application of Ranque-Hilsch Vortex Tube." *Int. J. Refrig.*, **49**, pp. 93-98 .
- [12] Wu, X. and Moin, P., 2009. "Direct Numerical Simulation of Turbulence in a Nominally Zero-Pressure-Gradient Flat-Plate Boundary Layer." *J. Fluid Mech.*, **630** : pp. 5–41.
- [13] Saffman, P. G., 1993. *Vortex Dynamics*. Cambridge Monographs on Mechanics, Cambridge University Press.

- [14] Förste, J., 1984. "Lugt, H. J., Vortex Flow in Nature and Technology. New York et al., John Wiley & Sons 1983. XV, 297 S., Zahlr. Abb., £ 47.45. ISBN 0-471-86925-2." *ZAMM - J. Appl. Math. Mech./Zeitschrift für Angewandte Mathematik und Mechanik*, **64**(11), p. 502 .
- [15] Jiang, M., Machiraju, R. and Thompson, D., 2005. "Detection and Visualization of Vortices." *Visualization Handbook*. Elsevier, Burlington, MA: pp. 295–309.
- [16] Robinson, S. K., 1991. "Coherent Motions in the Turbulent Boundary Layer." *Annu. Rev. Fluid Mech.*, **23**(1), pp. 601–639 .
- [17] Sadarjoen, I. A., Post, F. H., Ma, B., Banks, D. C. and Pagendarm, H. G., 1998. "Selective Visualization of Vortices in Hydrodynamic Flows,." *Proceedings of the IEEE Visualization Conference. Research Triangle Park, NC, Oct. 18-23, Vol 98*, pp. 419-422.
- [18] Haller, G., Hadjighasem, A., Farazmand, M. and Huhn, F., 2016. "Defining Coherent Vortices Objectively From the Vorticity." *J. Fluid Mech.*, **295**, pp. 136-173 .
- [19] Serra, M. and Haller, G., 2016. "Objective Eulerian Coherent Structures." *Chaos*, **26**(5), p. 053110 .
- [20] Zhang, Y., Liu, K., Xian, H. and Du, X., 2018. "A Review of Methods for Vortex Identification in Hydroturbines." *Renewable. Sustainable. Energy. Rev.*, **81**, pp. 1269-1285 .
- [21] Samtaney, R., Silver, D., Zabusky, N. and Cao, J., 1994. "Visualizing Features and Tracking Their Evolution." *Computer*, **27**(7), pp. 20-27 .
- [22] Hsu, William M, Corporation, Digital Equipment and Hunt, Neil, 1992. "SIGGRAPH '91 Workshop Report Integrating Computer Graphics, Computer Vision, and Image Processing in Scientific Applications Ingrid Carlbom , Digital Equipment Corporation Indranil Chakravarty , Schlumberger Laboratory for Computer Science." *Computer*, pp. 8-17.
- [23] Abdurakipov, S., Tokarev, M., Butakov, E. and Dulin, V., 2019. "Application of Computer Vision and Neural Network Analysis to Study the Structure and Dynamics of Turbulent Jets." *J. Phys. Conf. Series*, **1421**, p. 012018 .
- [24] Franz, K., Roscher, R., Milioto, A., Wenzel, S. and Kusche, J., 2018. "Ocean Eddy Identification and Tracking Using Neural Networks." *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*. Valencia, Spain, July 22-27.
- [25] Lguensat, R., Sun, M., Fablet, R., Mason, E., Tandeo, P. and Chen, G., 2018. "EddyNet: A Deep Neural Network for Pixel-Wise Classification of Oceanic Eddies." *International Geoscience and Remote Sensing Symposium (IGARSS)*. Valencia, Spain, July 22-27, pp. 1764-1767.
- [26] Deng, L., Wang, Y., Liu, Y., Wang, F., Li, S. and Liu, J., 2019. "A CNN-Based Vortex Identification Method." *J. Vis.*, **22**(1), pp. 65-78 .
- [27] Fukami, Kai, Fukagata, Koji and Taira, Kunihiro., 2020. "Assessment of supervised machine learning methods for fluid flows." *Theoretical and Computational Fluid Dynamics* Vol. 34 : pp. 497–519.
- [28] Brenner, MP, Eldredge, JD and Freund, JB, 2019. "Perspective on machine learning for advancing fluid mechanics." *Physical Review Fluids*, **4**(10), p. 100501 .
- [29] Brunton, Steven L, Noack, Bernd R and Koumoutsakos, Petros, 2020. "Machine learning for fluid mechanics." *Annual Review of Fluid Mechanics*, Vol. 52, pp. 477-508 .
- [30] Lee, Howon, Simone, Nicholas, Su, Yunxing, Zhu, Yuanhang, Ribeiro, Bernardo Luiz R, Franck, Jennifer A and Breuer, Kenneth., (2022). "Leading edge vortex formation and wake trajectory: Synthesizing measurements, analysis, and machine learning." *Physical Review Fluids.*, **7**(7), p. 074704 .
- [31] Ribeiro, Bernardo Luiz R and Franck, Jennifer A, 2023. "Machine Learning to Classify Vortex Wakes of Energy Harvesting Oscillating Foils." *AIAA Journal*, **61**(3), pp. 1281-1291 .
- [32] Cabral, B. and Leedom, L., 1993. "Imaging Vector Fields Using Line Integral Convolution." *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1993*. Anaheim, CA, Aug. 1-6, pp. 263-270.
- [33] Ni, Z., Chen, J., Sang, N., Gao, C. and Liu, L., 2018. "Light YOLO for High-Speed Gesture Recognition Key Laboratory of Ministry of Education for Image Processing and Intelligent Control, School of Automation, Huazhong University of Science and Technology, Wuhan 430074, China. NERCEL, Central China Normal,." *2018 25th IEEE International Conference on Image Processing (ICIP)*. Athens, Greece, Oct. 7-10, pp. 3099-3103.
- [34] Jiang, M., Hai, T., Pan, Z., Wang, H., Jia, Y. and Deng, C., 2019. "Multi-Agent Deep Reinforcement Learning for Multi-Object Tracker." *IEEE Access*, **7**, pp. 32400-32407 .
- [35] Farhadi, A. and Redmon, J., 2018. "Yolov3: An Incremental Improvement." *Computer Vision and Pattern Recognition*. Springer, Berlin/Heidelberg, Germany, pp. 1804-2767.
- [36] Wang, Q., Bi, S., Sun, M., Wang, Y., Wang, D. and Yang, S., 2018. "Deep Learning Approach to Peripheral Leukocyte Recognition." *PLoS One*, **14**(6), pp. 1-18 .
- [37] Calvet, Alejandro G, Dave, Mukul and Franck, Jennifer A, 2021. "Unsupervised clustering and performance prediction of vortex wakes from bio-inspired propulsors." *Bioinspiration Biomimetics*, **16**(4), p. 046015 .
- [38] Agatonovic-Kustrin, S. and Beresford, R., 2000. "Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research." *Journal of Pharmaceutical and Biomedical Analysis*, **22**(5), pp. 717-727 .
- [39] Yilmaz, Emre and German, Brian J., 2017. "A convolutional neural network approach to training predictors for airfoil performance." *18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, June. 18*, pp. 1-19.
- [40] Silva, Paulo., 2022. "Data supporting the publication 'Hovering rotor solutions by high-order methods on unstructured grids'."
- [41] Diosady, L. and Murman, S., 2015. "Case 3.3: Taylor Green Vortex Evolution." *Case Summary for 3rd International*

- Workshop on Higher-Order CFD Methods*. Orlando, FL, Jan 3-4.
- [42] Lesieur, M. and Métais, O., 1996. “New Trends in Large-Eddy Simulations of Turbulence.” *Annu. Rev. Fluid Mech.*, **28**(1), pp. 45-82 .
 - [43] Beck, A. D. and Gassner, G. J., 2012. “Numerical Simulation of the Taylor-Green Vortex at $Re = 1600$ With the Discontinuous Galerkin Spectral Element Method for Well-Resolved and Underresolved Scenarios.” *50th AIAA Aerospace Sciences Meeting*. Nashville, TN, Jan 9-12, pp. 1-4.
 - [44] DeBonis, J. R., 2013. “Solutions of the Taylor-Green Vortex Problem Using High-Resolution Explicit Finite Difference Methods.” *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition 2013*. Grapevine, TX, Jan. 7-10, pp. 1-20.
 - [45] Caradonna, F. X. and Tung, C., 1981. “Experimental and Analytical Studies of a Model Helicopter Rotor in Hover.” *Vertica*, **5**(2), pp. 149-161 .
 - [46] Ricci, Francesco, Silva, Paulo A.S.F., Tsoutsanis, Panagiotis and Antoniadis, Antonis F., 2020. “Hovering rotor solutions by high-order methods on unstructured grids.” *Aerospace Science and Technology*, Vol. 97, p. 105648 .



DSPACE

<https://dspace.org/>

A hybrid computer vision and machine learning approach for robust vortex core detection in fluid mechanics applications

Abolholl, Hazem Ashor Amran; Teschner, Tom-Robin; Moulitsas, Irene

2024-06-01

American Society of Mechanical Engineers (ASME)

<https://doi.org/10.1115/1.4064478>

Abolholl HA, Teschner TR, Moulitsas I. (2024) A hybrid computer vision and machine learning approach for robust vortex core detection in fluid mechanics applications. Journal of Computing and Information Science in Engineering, Volume 24, Issue 6, June 2024, Article number 061002