



# Wave Detection and Tracking Within a Rotating Detonation Engine Through Object Detection

Kristyn B. Johnson\*

National Energy Technology Laboratory, Morgantown, WV 26505

Donald H. Ferguson†

U.S. Department of Energy National Energy Technology Laboratory, Morgantown,  
West Virginia 26505

and

Andrew C. Nix‡ and Zachary Tallman§

West Virginia University, Morgantown, West Virginia 26505

<https://doi.org/10.2514/1.B38960>

As the operational time window of experimental rotating detonation engines (RDEs) is expanded and the technology matures toward integration within gas turbines, monitoring techniques must evolve to offer computationally efficient and highly time-resolved diagnostics. A computer vision object detection methodology that seeks to reduce data processing time and calculate wave velocity within drastically reduced time intervals as compared to traditional high-frame-rate RDE images analysis techniques is proposed. The adapted you-only-look-once object detection network is trained to detect individual detonation waves within single down-axis RDE images. The wave location and rotational direction detected within a frame are tracked through a series of high-speed images to calculate the frame-to-frame wave velocity with the time-step resolution of  $20\ \mu\text{s}$  across a series of frames. The analysis of the annotation box size and image linearization effects is presented, demonstrating the lowest frame-to-frame velocity total uncertainty of  $\pm 3.8\%$  and the highest classification speed of 9.5 frames per second using linearized images. Linearized images “unwrap” the RDE annulus pixel region to a reduced image size. This new method offers great reductions in data processing times and unsteady detonation behavior insight at intervals more comparable to the timescales of detonation wave interactions via the application of machine learning to experimental RDE data.

## Nomenclature

|                   |  |
|-------------------|--|
| $d_i$             | = linear distance between centroids in frames $i$ and $i + 1$ , pixels |
| $FR$              | = frame rate, frames/s   |
| $GT$              | = area of ground truth annotation, pixels                              |
| $i$               | = frame at current time step   |
| $i + 1$           | = frame at future time step  |
| $n$               | = number of data points, frames  |
| $P$               | = area of the predicted annotation, pixels                             |
| $R1$              | = rectangular annotation style 1                                       |
| $R2$              | = rectangular annotation style 2                                       |
| $r$               | = radius, m  |
| $r_{\text{RDE}}$  | = rotating detonation engine annulus radius, m                         |
| $r_{\text{Taub}}$ | = Taubin radius, pixels  |
| $\hat{r}_i$       | = Taubin center point, pixels  |
| $S1$              | = square annotation style 1  |
| $S2$              | = square annotation style 2  |
| $S3$              | = square annotation style 3  |

|                          |   |
|--------------------------|---|
| $s$                      | = arc length, rad                                     |
| $V$                      | = wave velocity, m/s                                  |
| $V_{\text{Bias}}$        | = wave velocity bias uncertainty, %                   |
| $V_{\text{Precision}}$   | = wave velocity precision uncertainty, %              |
| $V_{\text{Uncertainty}}$ | = wave velocity total uncertainty, %                  |
| $x_i$                    | = $x$ dimension of centroid in frame $I$ , pixels     |
| $x_{i+1}$                | = $x$ dimension of centroid in frame $i + 1$ , pixels |
| $x_{\text{Taub}}$        | = $x$ dimension of centroid in frame $i + 1$ , pixels |
| $y_i$                    | = $y$ dimension of centroid in frame $i$ , pixels     |
| $y_{i+1}$                | = $y$ dimension of centroid in frame $i + 1$ , pixels |
| $y_{\text{Taub}}$        | = $y$ dimension of centroid in frame $i + 1$ , pixels |
| $\Delta$                 | = annulus gap width, pixels                           |
| $\theta$                 | = angle of rotation between frames, rad               |
| $\Psi$                   | = noise reduction matrix                              |

## I. Introduction

PRESSURE gain combustion (PGC) offers an alternative strategy for advancing the thermodynamic efficiency of gas turbine engines. Historically, the approach to increased efficiency has relied on increasing turbine inlet temperatures. However, even with advanced cooling strategies, material limitations restrict maximum sustained temperatures. Conventional gas turbines are also limited by the occurrence of a pressure loss across the combustor. PGC, through technologies such as rotating detonation engines (RDEs), has the potential of increasing the pressure through detonation without the need for extracting additional work from the compressor. RDEs provide advantages to the turbine of not only the elevated pressure but also the same high temperatures permitted by advanced materials consistent with conventional gas turbine engines. Detonation waves travel at supersonic speeds around a cylindrical annulus, as shown in Fig. 1, consuming a detonable fuel–oxidizer mixture and creating a complex flowfield [1].

Presented as Paper 2021-1382 at AIAA SciTech, Virtual Event, January 11–15, 2021; received 7 July 2022; revision received 31 January 2023; accepted for publication 13 February 2023; published online 29 March 2023. Copyright © 2023 by the American Institute of Aeronautics and Astronautics, Inc. The U.S. Government has a royalty-free license to exercise all rights for Governmental purposes. All other rights are reserved by the copyright owner. All requests for copying and permission to reprint should be submitted to CCC at www.copyright.com; employ the eISSN 1533-3876 to initiate your request. See also AIAA Rights and Permissions www.aiaa.org/randp.

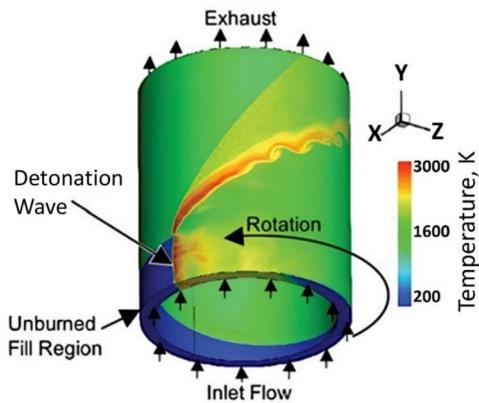
\*Research Mechanical Engineer, NETL Support Contractor; kristyn.johnson@netl.doe.gov. Member AIAA.

†Senior Research Engineer; donald.ferguson@netl.doe.gov. Member AIAA (Corresponding Author).

‡Associate Professor, Mechanical and Aerospace Engineering; andrew.nix@mail.wvu.edu. Member AIAA.

§Undergraduate Research Assistant, Mechanical and Aerospace Engineering; ztt0001@mix.wvu.edu. Student Member AIAA.

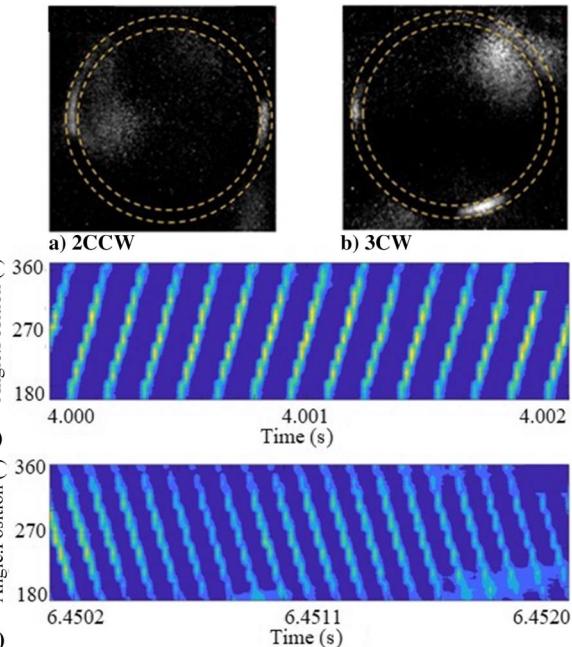
To capture and evaluate the unsteady detonation behavior within the RDE, high-speed image analysis techniques may be used. As RDEs move closer to an applied technology, it is expected there will be an increasing need for diagnostics that can resolve wave behaviors



**Fig. 1** RDE general characteristics. Adapted with permission from Nordeen et al. [1]. Copyright © 2011 by the AIAA.

with more refined time-step resolutions, or diagnostic outputs requiring less total data and demonstrating improved data processing times. Achieving modal classification and wave velocity with reduced data volume requirements will be necessary to achieve and optimize real-time RDE monitoring and control. High-speed imagery is one such diagnostic that typically requires post-test processing to provide useful information. However, using methods common in computer vision can assist with transitioning research imaging into an operational/production diagnostic. As volumes of recorded image data continue to grow, the benefits of incorporating machine learning capabilities into image analysis studies become more commonplace [2]. Using computer vision object detection, this study aims to train a neural network to detect, locate, and label individual detonation waves within an image according to their rotational direction. The coordinates of each wave can be tracked to evaluate individual wave speeds and frequencies in co- and counter-rotational modes.

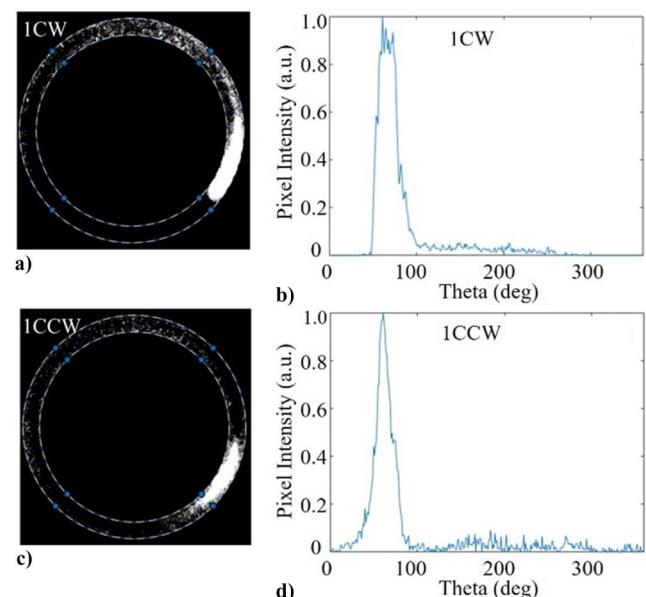
To observe the detonation wave behavior within an RDE, many optical techniques have been studied. Some optical studies rely on transverse images, which are captured perpendicular to the annulus center axis [3,4]. Down-axis images, defined as those captured downstream of the exhaust section and aligned with the center axis of the annulus, are commonly used to study wave propagation through time, which is done by concatenating polar pixel intensities to generate *detonation surfaces* or  $\theta - t$  plots [5–7]. Example down-axis images of two counterclockwise (2CCW) and three clockwise (3CW) wave modes are shown in Figs. 2a and 2b, respectively. Example detonation surfaces constructed using a series of such images are plotted in Figs. 2c and 2d, corresponding to 2CCW and 3CW wave modes, respectively. Detonation surfaces, while offering a unified and reliable means of modal classification and wave velocity calculation, require large image stacks to output behavioral diagnostics that are averaged over the imaging window. Most often, detonation surfaces are generated for 1000 images, providing velocity calculations with 20 ms time-step resolution, and assuming a frame rate of 50 kiloframes per second (kfps). Compared to the timescales of detonation wave interactions such as mode changes, galloping, and parasitic combustion, a 20 ms window is incredibly broad. Throughout the current study, 100 images are used to generate detonation surface plots to achieve a further reduced time-step resolution of 2 ms. Detonation surfaces provide time-averaged wave velocity data based on the fast Fourier transform (FFT) of a space-integrated matrix. Efforts to improve time step resolution by reducing the number of frames per velocity calculation reduces the resolution of the FFT, which analyzes frequencies of detonation waves completing multiple revolutions around the annulus, directly impacting the resolution of velocity outputs. In addition to the time-step resolution constraint of the detonation surface method, the method requires large proportions of computational time to obtain insight averaged over broad windows. However, the detonation surface is the most broadly used method in experimental RDE facilities due to the robustness and reliability of method. It is often considered the standard image processing technique, and it is therefore the baseline



**Fig. 2** Down-axis images showing a) 2CCW waves and b) 3CW waves, as well as detonation surfaces for c) 2CCW and d) 3CW wave modes across 100 images captured at 50 kfps [5].

for diagnostic improvement. The current method seeks to address both issues, which are the calculation time-step resolution and processing speeds, via the application of an object detection machine learning algorithm made viable by the availability of large data volumes.

In each of the mentioned studies, as well as the current work, images are captured with high frame rates at or above 50,000 frames per second (fps), or a 20  $\mu$ s time-step resolution. High frame rates result in an abundance of images, which require extensive processing times using conventional methods like detonation surfaces. Detonation behavior within these images is often apparent to the trained eye after filtering has been applied. For example, down-axis images in Figs. 3a and 3c show one clockwise (CW) wave and one counterclockwise (CCW) wave, respectively. The variations in pixel intensity through the centerline of each wave, shown in Figs. 3b and 3d, give evidence to the wave mode present. The discernable variation in



**Fig. 3** Down-axis images of a) 1CW and c) 1CCW detonation waves and their corresponding pixel intensities along the center radius of the annulus [1CW (Fig. 3b) and 1CCW (Fig. 3d)] [2] (a.u. indicates arbitrary units).

pixel intensity along different detonation modes as well as the abundance of available image data with known classifications are two factors supporting the application of computer vision to perform similar analyses.

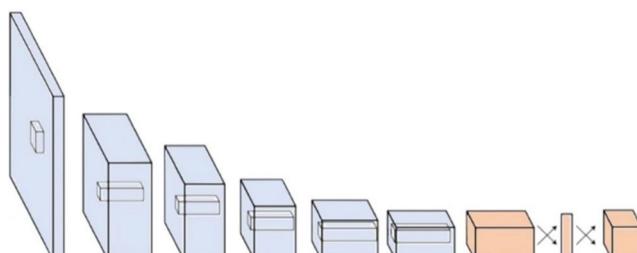
Computer vision, which is a field within artificial intelligence, aims to equip computers so that they may interpret and infer information from images similar to human capability. As a part of this broad effort, machine vision seeks to analyze images and videos to facilitate automated machine decisions based on image content. According to a 1980s-era description, “computer vision is the construction of explicit, meaningful descriptions of physical objects from images” and differs dramatically from conventional image processing [8]. Major improvements have been made in the field in recent years. Like many technological areas, improved computing capabilities have contributed to widespread advances. Computer vision, however, has specifically benefited from an exponential growth in available image datasets from smartphones, aerial and satellite imaging, video feeds, laboratory optical techniques, and many other sources. Applications of computer vision are becoming more commonly recognized by the public because adoption of the technology is widening across many industries such as autonomous vehicles, facial expression recognition, and three-dimensional medical imaging [9–11].

A common and widely developed capability within the field is image classification, which aims to predict a single label for a given image. Image classification capabilities have proven useful in a variety of engineering applications, most often through the use of neural networks. Directly applied to the combustion community, flames within a glass furnace can be classified to determine the number of active burners as well as the reactant flow rate [12]. In general, machine learning has improved modeling, diagnostics, and optimization of the combustion process in recent decades [13]. As an expected result of the sophisticated imaging techniques among RDE studies, image classification has proven to be a beneficial means of classifying the detonation wave number and direction using a convolutional neural network (CNN). In a study by Johnson et al. [2], a CNN was trained to classify 10 different wave modes within individual images at greater than 95% accuracy. Image classification, however, has the distinct disadvantage of only predicting a single label for each image. This becomes an issue when an image contains more than one object or a new arrangement of objects which are not included in the training set. Image segmentation is a possible means to address this issue, and it has been used previously to separate complex combustion patterns into constituent regimes [14]. In general, segmentation or clustering techniques partition a set of data into useful components, reducing data complexity and improving model accuracy [15]. The current work directly addresses the need for multiple labels and object location indicators through the use of a sophisticated computer vision technique of object detection.

It is important to distinguish object detection from the previously mentioned detonation surface method used widely throughout the RDE community. The baseline method of a detonation surface plot does not identify wave locations in individual frames. Instead, the method integrates pixel intensity within angular bins to visualize wave traces through time and space. Any valuable information regarding wave speed is gathered from frequency analysis of the detonation surface matrix spanning 100–1000 images, therefore reflecting time- and space-averaged information. Object detection attempts to locate observable objects within an image and constrain them with bounding boxes. Then, it identifies each detected object, which leads to more robust capabilities and the ability to discern information in images containing multiple objects. Object detection was shown to accurately locate key features of analog multimeters to automate readings from a live-video source [16], and it continues to offer valuable insight to document analysis, radar imaging, agriculture, and many other industries [17–19]. A family of model types that accomplishes this task is the region-based convolutional neural network (R-CNN), which scans an image to detect objects, and then uses a conventional CNN on each of those detected regions of interest (ROI) to predict the label of each object [20]. The use of a CNN for each identified object introduced an issue of efficiency, which was

later addressed with the development of the fast R-CNN, which is a new model type that performs feature extraction before detecting ROIs [21]. Shortly after the development of the fast R-CNN, the faster R-CNN was introduced and showed higher accuracy and speed performance than its predecessors. Faster R-CNNs outpaced those predecessors by using a pretrained CNN to propose ROIs and the objects within the ROIs, which were then fed to a fast R-CNN [22].

A later type of object detection model, referred to as the you-only-look-once (YOLO) model, was created to specifically enhance object detection speeds. Unlike other object detection methods, which use sliding windows or region proposal methods to first detect potential bounding boxes and then run regions through separately trained classifiers, YOLO achieves drastically improved detection speeds by performing unified detection and labeling entirely with one neural network. The YOLO architecture, shown in Fig. 4 [23], relies on 24 convolutional layers feeding to two fully connected layers. Unified detection considers object presence in relation to the global image information. The YOLO network first parses an image into a grid. Each cell grid is used to separately predict possible bounding boxes and conditional class probability. The network's confidence in the size and location of a bounding box for which the centroid lies within a given grid cell is multiplied by the class confidence of that same grid cell, resulting in final class-specific confidence scores. Final detections with class-specific confidence scores falling above the user-defined confidence threshold are output as object detection annotations. The effects of confidence thresholding will be discussed in a later section [23]. The ability to perform all necessary tasks within a single network pushes the YOLO architecture toward real-time object detection capabilities. It is important to note that although YOLO is often considered a solution for real-time object detection, the processing speeds are adequate for standard streaming video, which is orders of magnitude slower than imaging speeds within the RDE community, as well as the timescale of the detonation wave progression. For this reason, the RDE YOLO network proposed within this work cannot accomplish real-time RDE monitoring within the existing experimental setup but remains beneficial as a unique means to calculate frame-to-frame detonation wave velocities with a time-step resolution of 20  $\mu$ s by tracking YOLO annotation coordinates through successive images. As improved versions of the YOLO framework are continually released, improved processing speeds will be attainable through replication of the methodology described here. In addition to further improved processing speeds, an improved experimental setup where a short burst of high-speed images can be captured and analyzed during RDE operation may also extend the method to real-time application. The network does so at improved data processing speeds as compared to widely used techniques that offer only average velocities across a considerably larger operational window: usually 20 ms. Compared to other object detection methods, YOLO learns representations of objects that are more generalizable due to the contextual consideration of the entire image, making it less likely to fail if applied to varied down-axis RDE imaging techniques across multiple laboratories. To evaluate this capacity, images captured in an external RDE laboratory, as well as images captured with different camera orientations and RDE geometries, are evaluated in a later section.



**Fig. 4** YOLO network architecture [23] with convolutional layers indicated in blue and connection layers indicated in orange. Adapted with permission from Redmon et al. [23]. Copyright © 2016 by the Institute of Electrical and Electronics Engineers.

## II. Experimental Setup

Data used in this study were collected from experiments using an RDE with a ducted exhaust, which is fully integrated with a downstream backpressure control valve to emulate downstream turbomachinery, permitting control of the precombustion pressure partially independent of the exit geometry of the RDE [2]. A photograph of the rig is shown in Fig. 5a, highlighting the combustor, postcombustor, and exhaust sections. Shown as a cross section in Fig. 5b, the RDE design features an annulus with a mean diameter of 152 mm, a length of 100 mm, and a gap width of 5 mm. The device enables the continuous detonation of radially injected air and fuel mixtures, which are composed of diatomic Hydrogen ( $H_2$ ) and natural gas. Instead of the more common use of a predetonator, the mixture is ignited by an afterburner located in the exhaust section. Because the RDE is uncooled, testing is typically limited to 6–10 s to prevent damage to the inner- and outer-body hardware as a result of incredibly high temperatures associated with detonation (greater than 1700 K).

Pressure, Hydroxyl radicals ( $OH^*$ ) chemiluminescence, and flow ionization (combustion ionization) data are acquired at 1 MHz using a National Instruments PXIe chassis. Down-axis images are captured through a 50-mm-diameter quartz viewport at 50,000 fps by an imaging system that consists of a Nikkor 105 mm UV lens mounted to a Vide Scope VS4-1845HS-UV intensifier and a Photron FASTCAM SA-Z high-speed digital camera. Although the high-speed camera's frame-rate capabilities reach 1 megaframe per second, the reduced frame rate offers a desirable image resolution of  $512 \times 512$  pixels. The imaging system relies on a 45 deg mirror positioned perpendicular to the RDE axis to avoid hardware damage in the unlikely event of viewport failure.

## III. Methodology

The process presented in this section includes three distinct efforts. The first effort is dedicated to the creation and treatment of an image dataset spanning a variety of wave modes and operating conditions. The second major effort develops and trains a neural network based on the YOLO family, which can detect and classify each individual detonation wave within an image. The third and final effort processes the output of the network to determine wave mode, velocity, and frequency. It should be noted that steps 1 and 2 only need to be performed once because the trained neural network would be applied to future datasets without the need for additional training, thus providing a rapid means of quantifying detonation wave performance.

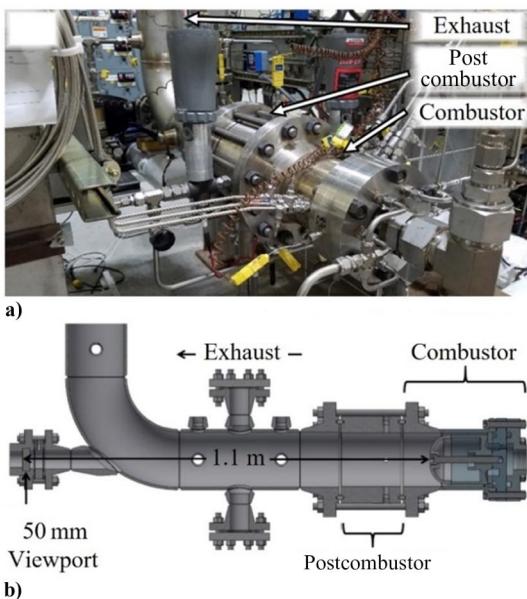


Fig. 5 Representations of a) RDE facility and b) RDE cross section.

### A. Image Treatment

To adequately train an object detection network, a dataset of images must be selected. To prevent overfitting to features that can be associated with artifacts of specific test dates or wave modes, a variety in images is a necessity. The image set created for this study includes 500 images that were captured across various dates, operating conditions, and multiple experimental facilities [2,24–26]. Although 500 images may seem like a low threshold of data volume as compared to what is required by other artificial intelligence techniques, many images containing multiple detonation waves will provide multiple annotations in the Network Development section (Sec. III.B).

To prepare the image set for training, raw images are preprocessed identically. An example raw image is shown in Fig. 6a. A short series of images is first used to generate an average image, shown in Fig. 6b. From the distribution of pixel intensities in the average image, a Taubin fit is used to algebraically fit a circle to the data [27]. The circle is inherently associated with the mean radius of the RDE annulus and is used to crop each image to a reduced size of  $301 \times 301$  pixels centered about the annulus. Once cropped, each image is multiplied piecewise by a scaled, noise reduction matrix  $\Psi$  to reduce the pixel magnitudes outside of the area of interest. The noise reduction matrix is created using Eq. (1) in polar coordinates, where  $r_{Taub}$  is the annulus radius found using the Taubin fit and  $\Delta$  is the annulus gap width in pixels. For the current image set,  $r_{Taub}$  and  $\Delta$  were found to be 130 and 20 pixels, respectively. The contour of the resultant noise reduction matrix  $\Psi$  is shown in Fig. 6c, with the dark to light contour corresponding to values from zero to one.

Once pixel intensities have been reduced outside the region of interest, images are filtered with a basic contrast adjustment that linearly maps pixel intensities such that the lowest and highest 1% of all pixel values are saturated. The resultant filtered example image is shown in Fig. 6d. Each step of the outlined image treatment process requires minimal computational effort. However, this treatment sufficiently prepares each image for feature extraction within the neural network:

$$\Psi(r, \theta) = \begin{cases} \left(\frac{r}{r_{Taub}}\right), & r \leq r_{Taub} - \frac{\Delta}{2} \\ \left(\frac{r_{Taub}}{r}\right), & r \geq r_{Taub} + \frac{\Delta}{2} \\ 1, & r_{Taub} - \frac{\Delta}{2} < r < r_{Taub} + \frac{\Delta}{2} \end{cases} \quad (1)$$

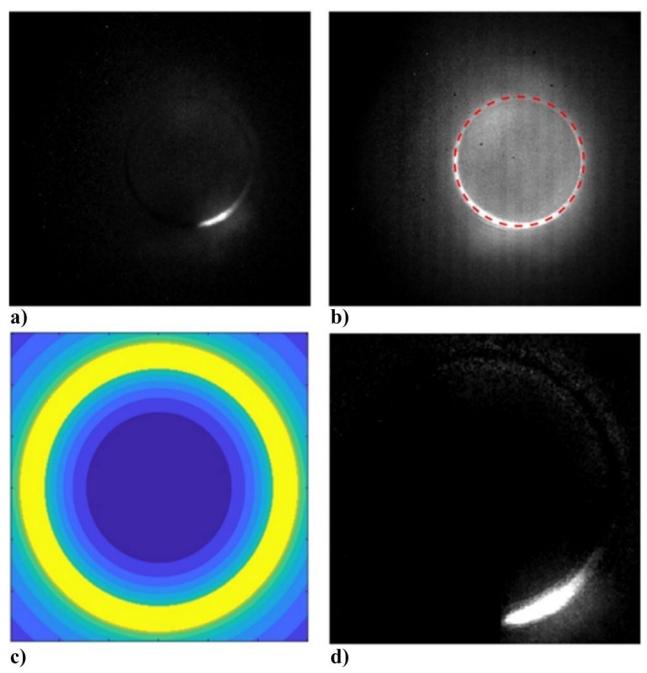


Fig. 6 Image treatment process: a) example raw image, b) average image with Taubin fit circle overlaid, c) noise reduction matrix  $\Psi$  contour plot, and d) final cropped and filtered example image.

## B. Network Development

As mentioned in an earlier section, object detection varies significantly from image classification. However, the phases of development are very similar for both techniques due to their shared reliance on CNNs, leading to a methodology flowchart similar to those used in previous studies [2]. However, differences among substeps should be recognized and regarded with great significance. The current method, outlined in Fig. 7, includes four major phases: imaging, YOLO CNN training, YOLO CNN validation, and inference.

### 1. Imaging Phase

To begin the imaging phase, images from the created dataset should be manually annotated. Manually annotating images, which is arguably the most crucial step of this process, includes visually identifying each detonation wave present in an image, creating a bounding box around each ROI, and labeling each bounding box according to wave direction. The process of annotation can be accomplished using a custom annotation tool, or one of several publicly available annotation platforms. The current study uses Microsoft's VoTT (which stands for visual object tagging tool) application [28] to manually annotate wave fronts with rectangular bounding boxes and appropriate labels or tags. For each completed image, VoTT exports coordinate pairs for the corners of each bounding box along with the designated label to a comma-separated values (.csv) file.

Object detection models are commonly configured to detect as many as 20–80 object category labels, such as a plane or a dog, as required by popular competitions [29,30]. Annotations for this study, however, will be labeled as one of only two object categories: CW or CCW detonation waves. Examples of wave annotations are shown in Fig. 8; annotations of the CW waves are shown with green bounding boxes, whereas those of the CCW waves are shown with orange bounding boxes. To again highlight the major difference between this work and the previous study [2], the predicted *image classification labels* that include the wave number as well as the direction of rotation are shown in the top left corner of each example image. The image classification labels differ drastically from the bounding boxes predicted by the current object detection method. In Figs. 8a and 8b, the image classification labels are 1CCW and 2CW, respectively. The annotation boxes still inform the wave number and direction while providing additional information regarding the location of each wave. The previous image classification network would identify and assign a single label to Figs. 8c and 8d, containing counter-rotating (CR) detonation waves, whereas object detection identifies each detonation wave and its specific rotational direction [2]. The ability to properly identify and analyze CR wave modes (occurring when one or more pairs of CW and CCW waves are present) is critical to the previous and the current works. All wave modes in the studied RDE are a variation of only the CW waves, the CCW waves, or both to create a CR mode. Therefore, the use of bounding boxes also results in the ability to properly analyze wave modes not present in the training set, which was otherwise not possible with image classification. It is important to note that the annotations must be performed manually to generate an annotated image set, which is then split into training and validation image sets.

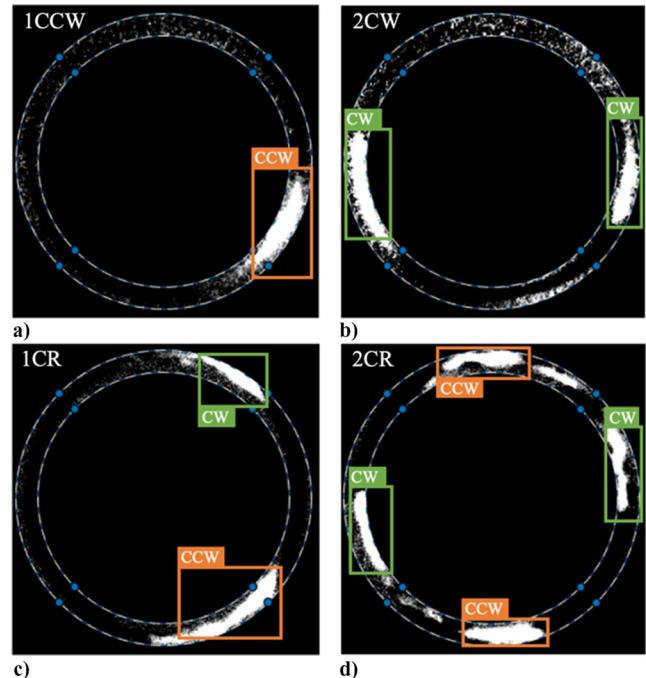


Fig. 8 Annotation of RDE down-axis images of a) 1CCW, b) 2CW, c) 1CR, and d) 2CR modes [2,25].

### 2. CNN Training Phase

Once split into respective image sets, the YOLO CNN training phase may begin. As is common with most applications of neural networks, training is initialized with pretrained weights that are tuned to the dataset of interest. The pretrained weights used in this study are associated with Darknet: an open-source feature extraction framework [24]. Using pretrained weights, which is a common form of transfer learning, shortens the training process by initializing weights that have previously trained on similar object detection tasks. To train the network for this specific problem, images and their annotations are first fed to the initialized YOLO network. As the network repeatedly evaluates the entire image set, referred to as an epoch, parametrized weights are internally updated. At each update, predicted annotation locations, sizes, and wave classifications are compared to values associated with manual annotations. Unlike the previous study, a standard accuracy measurement is not possible because the size and location of the YOLO prediction must be compared. To evaluate both outputs, the intersection over union (IOU) is calculated for each predicted annotation. The IOU metric is widely used in object detection [31] and is calculated using Eq. (2), where  $GT_i$  represents the area of the ground truth annotation and  $P_i$  represents the area of the predicted annotation. The spatial significance of IOU, or Eq. (2), is illustrated in Fig. 9. Although the IOU is calculated for each annotation box, an overall measurement of performance is calculated at each epoch using the multipart YOLO loss function [24]. The YOLO loss function is a sum-squared error approach, accounting for classification loss, localization loss, and confidence

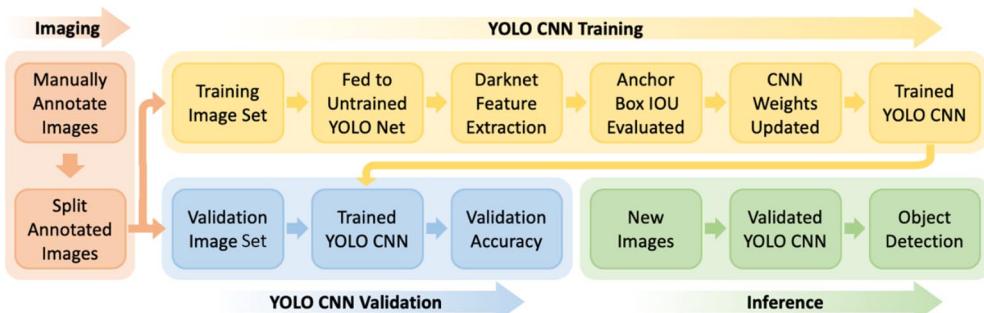


Fig. 7 YOLO CNN development method flowchart.

loss. Each component will be further addressed in a later section. Once the performance meets a desired threshold (specifically when the validation loss does not improve over 10 epochs), the weights are finalized and the network is deemed a trained network:

$$\text{IOU} = \frac{GT_i \cap P_i}{GT_i \cup P_i} \quad (2)$$

### 3. CNN Validation Phase

Once training of an epoch is complete, the YOLO CNN validation phase can be performed at each epoch concurrently. To validate the model, the validation image set is fed to the trained YOLO CNN without annotations. The trained CNN is used to perform object detection; again, the loss and the IOU of the predicted anchor box locations, sizes, and object classifications can be evaluated. If performance exceeds the desired threshold, the trained network is then considered to be a trained and validated YOLO network.

### 4. Inference

From the point of network validation, the trained network is able to call on finalized weights, and is therefore available to detect CW and CCW waves and their respective locations in new laboratory images. That is to say, the phase of inference is treated as a tool that can be used indefinitely, although detection is limited by training; for example, a network trained on detonation objects could not predict the presence of deflagration. Detections output dimension clusters, including four coordinates for each anchor box as well as the object classification and the network confidence of prediction. This is unique to the current object detection method, where the previous image classification method only output a single classification of the entire image. Although the network is finalized, the inference technique is further refined in two distinct ways, with both ensuring the output of the network meets the determined standard required by the current study. The first alteration increases the threshold of confidence required for a predicted annotation to be reported. The second refinement calls on the IOU formulation to enable nonmaximum suppression.

Nonmaximum suppression is a computer vision technique used to prevent the reporting of multiple, overlapping bounding boxes for a single object [32]. In simplest terms, when the network attempts to predict two annotations of the same label that are overlapping, the IOU of the two boxes is calculated; and when that IOU exceeds a chosen threshold, the annotation with the lower confidence is discarded. Due to the physical understanding of detonation waves in the RDE, the improbability of two waves of the same rotational direction overlapping in an image allows the nonmaximum IOU threshold to be significantly increased. Under these conditions, anchor boxes predicted by the proposed object detection method can be used in the following effort of wave speed calculation.

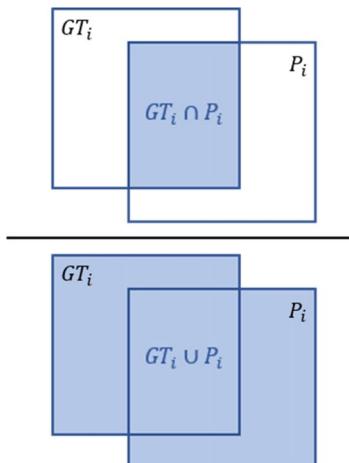


Fig. 9 Visualization of intersection over union.

### C. Network Output Processing

Once objects are located and the number of waves and wave modes are determined, postprocessing can be performed to determine individual wave speed. The trained YOLO network outputs two coordinate pairs for each detected bounding box, indicating the minimum and maximum  $x$  and  $y$  coordinates. The centroid coordinate location of each annotation box is easily found as the midpoint of the two provided coordinate pairs. The four coordinate values are concatenated with the image file name, the image count within the series, the object label, and the confidence score as a single vector. Each detection vector is stacked throughout the inference stage to create an output summary matrix. Therefore, the analysis of an image with  $N$  waves would result in  $N$  detection vectors appended to the summary matrix. In other words, if 50 images depicting a two-wave mode were perfectly annotated, the summary matrix would contain 100 rows of data. Velocity calculations can be performed after each image is processed, when considering newly appended detection vectors, or after a series of images are all processed, analyzing the complete summary matrix. The progression of each detonation wave can be tracked from image to image using the dimension cluster output of the inference phase of the YOLO CNN. Although the network will not associate objects from one frame to the next, high frame rates ensure that wave progressions between frames remain minimal. Therefore, a simple assumption can be made such that when multiple waves are detected, each wave corresponds to the wave in the next frame that is closest to the original position. Because the waves detected are also classified as CW or CCW, counter-rotating waves are less likely to be incorrectly labeled. Additionally, CW and CCW waves within a counter-rotating wave pair can be tracked independently to compare velocity variations between oppositely rotating wave groupings. In each case, the progression of each wave can be compared to the frame rate and the nominal annulus diameter to determine wave velocity. The process is initiated by first finding the linear distance  $d_i$  between the centroids of two bounding boxes in successive frames using Eq. (3), where the annotation centroids in frames  $i$  and  $i + 1$  are located at coordinates  $(x_i, y_i)$  and  $(x_{i+1}, y_{i+1})$ , respectively. Assuming both centroids lie along the annulus circle with calculated radius  $r_{\text{Taub}}$ , the angle of rotation between frames  $\theta_i$  can be calculated using Eq. (4). To finally calculate the velocity of the wave, the true distance traveled along the circumference of the annulus must be determined according to the geometric relations outlined in Fig. 10.

By relating the angle of rotation and the annulus physical radius  $r_{\text{RDE}}$  to the traveled arc length  $\Delta s$ , Eq. (5) can be used to calculate the wave velocity  $V_i$  for a given frame. Additionally, successive frames can be used to correct sparse errors of the detected wave direction:

$$d_i = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (3)$$

$$\theta_i = 2\sin^{-1}\left(\frac{d_i}{2r_{\text{Taub}}}\right) \quad (4)$$

$$V_i = \Delta s / FR = r_{\text{RDE}} \cdot \theta_i / FR \quad (5)$$

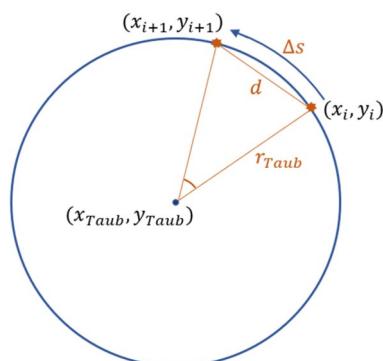


Fig. 10 Geometric relationship leveraged for wave velocity calculation.

## IV. Results

### A. Baseline Network Performance

To prepare for the training of the network, an image set of 500 images was manually annotated. Although this is a relatively small collection of images as compared to the data volume collected from the high-speed camera, the costs of manual annotation were minimized to that required to successfully train the networks. Among the 500 images, a variety of wave modes, dates, and operating conditions were covered. Due to multiple waves being present in many images, a total of 771 annotation boxes were manually created and labeled. Those annotations were converted to a text-based YOLO format that calls on the specified image file path and associates that image with the dimension cluster of each annotation. With the inputs available, the network can begin to train the parametrized weights. As mentioned in a previous section, the loss of the updated network is evaluated and recorded after each epoch. The behavior of the loss reduction over the process of training is shown in Fig. 11. The YOLO loss function represents the sum-squared error of classification loss, localization (coordinate) loss, and confidence loss between the incremental network output and the ground truth data. Classification loss is the squared error of class conditional probabilities for each class, only contributing to total loss if an object is detected. Localization loss, or coordinate loss, is a measure of the difference in the ground truth annotation size and the location to those of the predicted bounding boxes. Finally, confidence loss is associated with the probability of an object or, in this case, a detonation wave being present in a given bounding box, indicating the *objectness* of the region. The encompassing YOLO loss function is assessed for the network's evaluation of both training and validation sets at each epoch in Fig. 11.

In this figure, the training loss and validation loss are both shown to decrease considerably in the first 20 epochs, which is expected as the network begins to understand features extracted from the image set. The initially decreasing loss is followed by a near-plateau around 50 epochs, and then a sharp decrease is experienced in both trends at the 52nd epoch. The major and instantaneous improvement in loss is due to a significant reduction in batch size from 32 to four images. Neural networks attempt to minimize a loss function of the training parameters. At a smaller batch size, the gradient of the loss function is updated using fewer images. Therefore, model parameters are updated in smaller "steps" toward the loss minimum, resulting in the refinement of the network over the final 48 epochs. Larger batch sizes are used in the early epochs as an effort to avoid converging to local minima while also optimizing the computational budgeting. Unlike a percent accuracy measurement, there is no given or accepted threshold of loss. Instead, its slope is used as a measure of progressive learning, informing reduced learning rates, and early stopping to prevent overfitting of the model.

Once the network weights are finalized, the network is available for inference efforts. However, as mentioned earlier, the confidence and IOU thresholds must be altered to ensure proper results and to

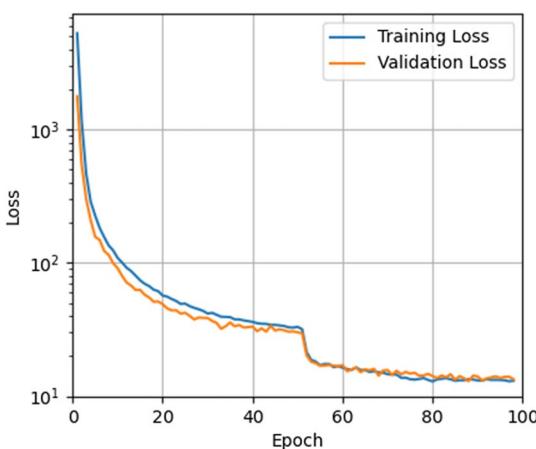


Fig. 11 YOLO training and validation loss across 100 epochs.

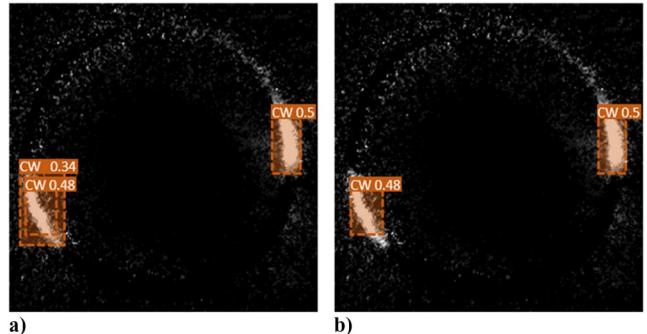


Fig. 12 Results of nonmaximum suppression: a) example image before IOU thresholding, and b) example image after IOU thresholding.

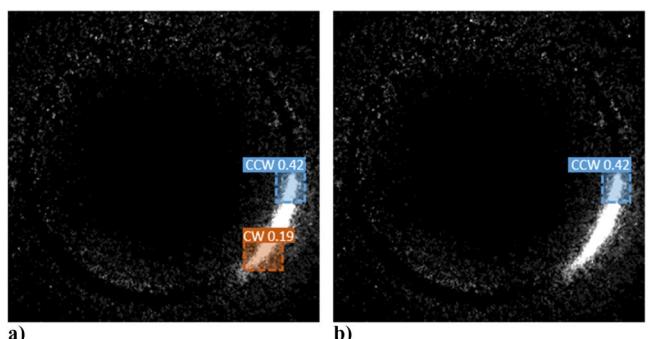


Fig. 13 Results of confidence thresholding: a) example before confidence thresholding, and b) example after confidence thresholding.

perform nonmaximum suppression. Appropriately altering each threshold is done by simply feeding new, unseen images to the trained YOLO network; performing inference at low thresholds of confidence and IOU; and increasing each threshold independently until the results are satisfactory. Examples of the resulting improvement are shown in Figs. 12 and 13, where nonmaximum suppression and confidence thresholding have direct impacts on the network output, respectively. Specifically, Fig. 12 shows an example image where two CW waves are present but three bounding boxes are predicted. It can be seen that two of the predicted bounding boxes are overlapping on the same wave, and can therefore be resolved by increasing the IOU threshold. Once the IOU threshold is increased to a value that encompasses the scenario of these two overlapping boxes, the box with the lower confidence score (displayed to the right of each classification) is disregarded, resulting in a more accurate output, shown in Fig. 12b. Figure 13 displays an instance where confidence thresholding plays a major role in network output. In Fig. 13a, an image containing one CCW wave is present, but an additional CW bounding box is predicted. The confidence of the incorrect prediction corresponds to a very low confidence value of 0.19 and can be disregarded by simply increasing the confidence threshold value. The optimal values of the confidence and IOU thresholding vary between image types and network iterations but are easily altered with minimal effort.

### B. Annotation Size Refinement

Having demonstrated that an object detection network can be trained to accurately locate and label individual detonation waves, it is possible to further refine the network for improved end usage. Because velocity calculations are the end goal of this study, it is beneficial to study the impact of annotation geometries. As with any supervised learning endeavor, the output of the network inherits the form of the input created by the practitioner. Therefore, the variation in the annotation size and location associated with the training dataset will be directly reflected in the network predictions, which has direct implications on the upcoming velocity calculation. Specifically, irregularly sized and larger annotation boxes are found to create

larger errors in velocity calculations, whereas smaller, uniform bounding boxes result in reduced classification accuracy of the rotational direction. To form a more encompassing understanding of these relationships, multiple networks were trained with various annotation styles. Examples of each style are depicted on the sample images in Fig. 14.

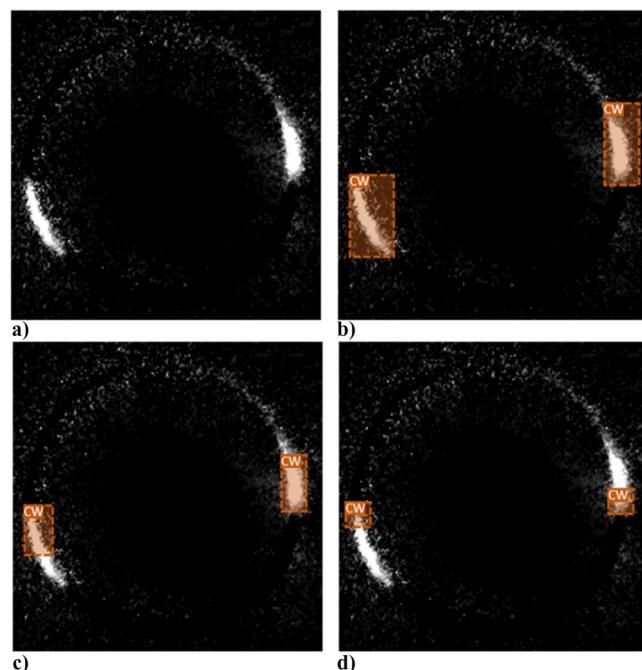
The example image (Fig. 14a) shows two clockwise waves. The initial annotation style (Fig. 14b) encompasses the entire detonation wave, and it has a strong ability to properly identify the wave direction. The second iteration (Fig. 14c) attempts to reduce the average annotation area to refine velocity calculations. However, simply reducing the average annotation area did not fully resolve the issue of large velocity fluctuations because the centroid location relative to the wave front remains inconsistent from frame to frame. The final iteration of the annotation style (Fig. 14d) uses small square annotation boxes consistently centered about a location just behind the wave front. A consistent relationship between the wave profile geometry and the annotation centroid results in a more accurate and stable velocity calculation.

For variation of each annotation style shown in Fig. 14, a network is trained according to the proposed methodology. To better evaluate the expected improvements in velocity calculations resulting from varying annotation styles, each network was used to perform predictions on a test image set. The centroids of the annotation boxes were then used to generate a best-fit circle using the Taubin fit algorithm. Finally, the root-mean-square error (RMSE) is calculated for each centroid radius  $\hat{r}_i$  relative to the Taubin center point  $(x_{\text{Taub}}, y_{\text{Taub}})$  and the Taubin radius  $r_{\text{Taub}}$  using Eqs. (6) and (7). As a general rule, as the Taubin RMSE decreases, the reliability of a subsequent velocity calculation improves:

$$\hat{r}_i = \sqrt{(x_i - x_{\text{Taub}})^2 + (y_i - y_{\text{Taub}})^2} \quad (6)$$

$$\text{RMSE} = \sqrt{\sum_{i=1}^n \frac{(\hat{r}_i - r_{\text{Taub}})^2}{n}} \quad (7)$$

The results of five networks trained on varying annotation styles are recorded in Table 1 according to the annotation style (R1, R2, S1, S2, and S3), the average annotation area, and the Taubin RMSE.



**Fig. 14** Progression of manual annotation variations on an image containing two CW waves: a) baseline image, b) annotation style R1, c) annotation style R2, and d) annotation style S3.

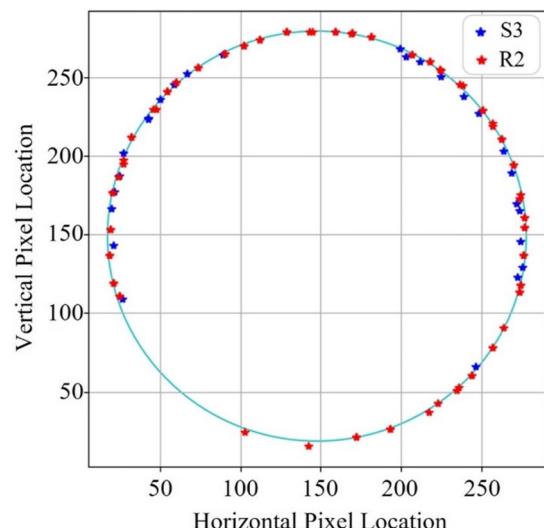
**Table 1** Taubin RMSEs for varying annotation styles and areas

| Annotation style | Area, pixels | Taubin RMSE      |
|------------------|--------------|------------------|
| Rectangular (R1) | 5070         | 3.391            |
| Rectangular (R2) | 809          | 2.311            |
| Square (S1)      | 425          | 0.973            |
| Square (S2)      | 318          | 0.907            |
| Square (S3)      | 213          | 0.886            |
| Linearized (lin) | 320          | N/A <sup>a</sup> |

<sup>a</sup>N/A denotes “not applicable.”

An additional annotation style, called lin (for linearized), is also shown in Table 1; it will be detailed in the next subsection. The lin annotation corresponds to images that reduce the image size by linearizing (or unwrapping) the annulus pixel region. It is shown that the first rectangular style (R1), which corresponds to the example shown in Fig. 14b, results in the largest Taubin RMSE. Reducing the area but maintaining the rectangular style, which corresponds to Fig. 14c, significantly reduces the experienced Taubin RMSE. In an effort to further reduce the Taubin RMSE, three variants of the square annotation style were used, corresponding to Fig. 14d. Networks trained with square annotations reported lower values of the Taubin RMSE, insinuating a lower variation in wave speed calculations. Generally speaking, a trend is shown that as annotation area decreases, the Taubin RMSE is improved. The centroids predicted by the networks trained using the R2 and S3 styles are plotted in Fig. 15, overlaid on the corresponding Taubin fit circle. The figure highlights the consistency of predictions in relation to how tightly they fall along a given circumference for each annotation style. Although both datasets seem to tightly fit the Taubin circle, centroids associated with R2 show slightly more variance, contributing to the higher RMSE value shown in Table 1. Note that in previous studies [2,3], it was documented that in the experimental setup, a portion of the down-axis view (approximately 60 deg) of the RDE was partially obstructed. As shown in Fig. 15, in this region, detonation waves are not imaged, and therefore not well detected.

Unfortunately, an expected fault of smaller square annotations is a loss of classification accuracy. By further inspecting the example annotation areas, Fig. 16 shows the inherent variation of information contained within annotations of different sizes. As the pixel area within annotations decreases, so do the contained features that can be extracted, and therefore the network’s ability to distinguish between CW and CCW waves. The resulting conclusion is that each annotation style is beneficial in differing metrics. If wave direction classification is of the most importance to a given study, the larger rectangular style (R1) is best suited. On the other hand, if wave



**Fig. 15** Annotation centroids predicted by annotation styles R2 and S3 and the corresponding Taubin fit circle.

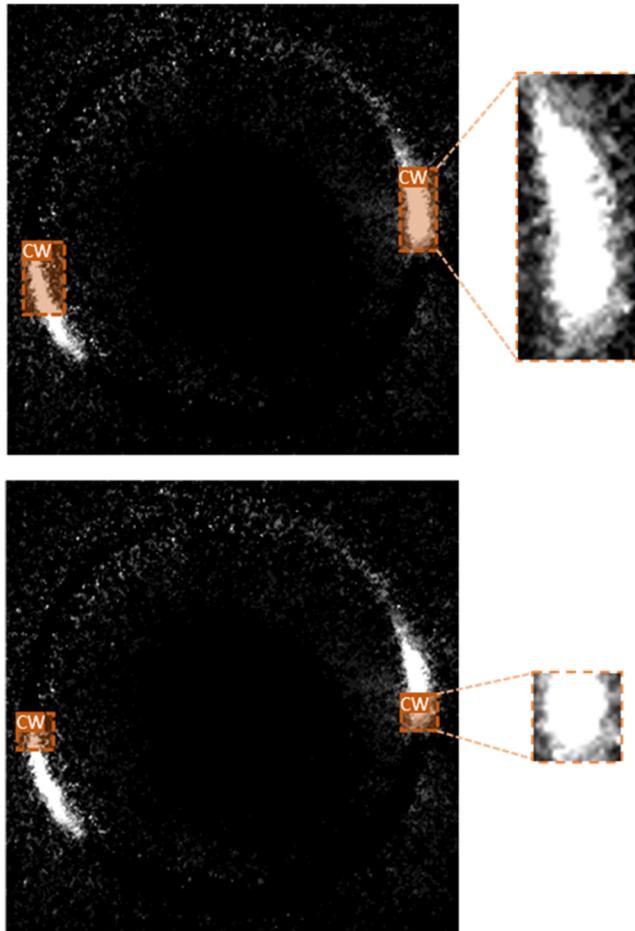


Fig. 16 Highlighted features associated with varying annotation box areas.

velocity is of the most importance to the user, a small square size such as S2 or S3 would be preferable. Because the final portion of this study is concerned with velocity calculations, and because determination of the wave direction is still possible in postprocessing, the model trained using style S3 will be used to generate wave velocity results for the remainder of the text with comparison to the R2 style.

### C. Consideration of Linearized Network

In an effort to address the errors associated with annotation coordinates and the subsequent Taubin fit error, a final variation of the YOLO network is trained for linearized (or unwrapped) images. Linearized images eliminate unnecessary pixels outside of the RDE annulus, reducing the total pixels passed to the network, and therefore further improve processing times. In general, square images are sized  $301 \times 301$  pixels, whereas linearized pixels are sized  $32 \times 736$ , leading to a 74% decrease in the total pixel area. To create the linearized images, pixels are extracted at radius locations spanning  $\pm 16$  pixels from the annulus radius (stated previously as 130 pixels) and at 736 azimuthal locations. An example image is displayed with azimuthal lines and the S3 annotation overlaid in Fig. 17, with the image zoomed in to the lower-right quadrant to visualize the heavily condensed azimuthal lines. By stacking the pixels along each line, shown in cyan, the annulus is unwrapped to a resultant format with the  $x$  and  $y$  axes corresponding to  $\theta$  and  $(\hat{r}_i - r_{\text{Taub}})$ , respectively.

Unfortunately, the linearization of a hollowed cylindrical region, such as the RDE annulus region, results in lost pixels. Pixel loss associated with linearization increases with increased inner and outer radii, and decreased angular sampling. In the current format, 9.9% of the annulus region pixel area is forfeited. Although selection of a sparser angular sampling such as 360 lines would further reduce the input image size, it would also result in 55.9% annulus region pixel area loss, which would likely be detrimental to wave feature

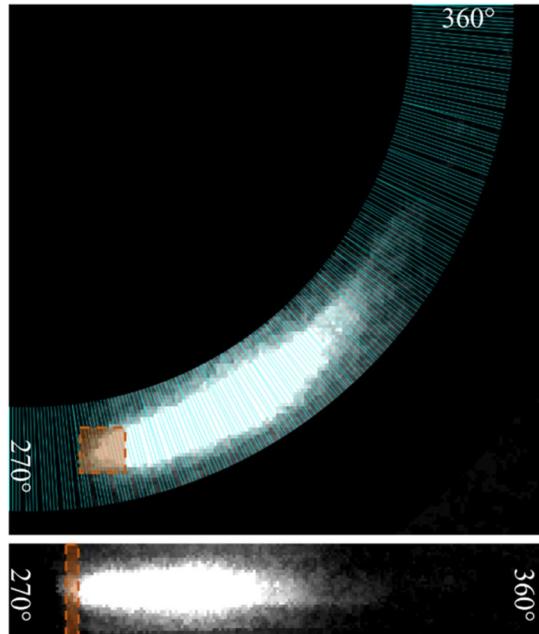


Fig. 17 Example quadrant unwrapped to linearized image with translated annotation box.

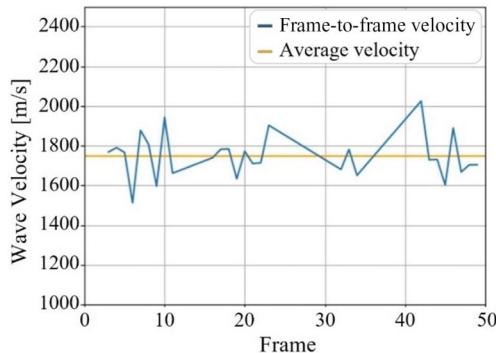
preservation. The example quadrant is linearized in the lower portion of Fig. 17, showing the conservation of wave features despite the pixel loss. Fully linearized images will be shown in sequence in a later section. The new annulus orientation enables an annotation style that reflects only the angular location of the wave, simplifying velocity calculations by circumnavigating Eqs. (3) and (4). Instead of determining the location of detonation waves in two spatial dimensions, the lin aims to determine only the angular “bin” in which a wave front resides. Lin annotations span the full radius range and are consistently 10 pixels in width, corresponding to a width of 4.905 deg (0.0856 rad) and an annotation area of 320 pixels. The width of 10 pixels is centered about the wave front to foster recognition of the profile edge. A comparison of an S3 and a lin annotation is displayed in Fig. 17. The linearized images are analyzed by an unaltered, flexible YOLO network employed consistently throughout the entire study, requiring only a minimum dimension of 32 pixels for filter compatibility. Although it is possible to extract a 1-D trace along the Taubin radius, spatial features of the wave front would be lost, possibly leading to poor directional confidence and depreciation of velocity calculations. Additionally, object detection for the 1-D signal would require a drastically different network architecture, differing entirely from YOLO, which does not align with the scope of the current study. Still yet, a machine learning technique applied to a 1-D signal, either extracted at the Taubin radius or averaged across the annulus width, would likely offer reduced processing times due to lower data dimensionality and volume. Therefore, the consideration of object detection from a 1-D trace is reserved as a future work.

### D. Wave Speed Calculations

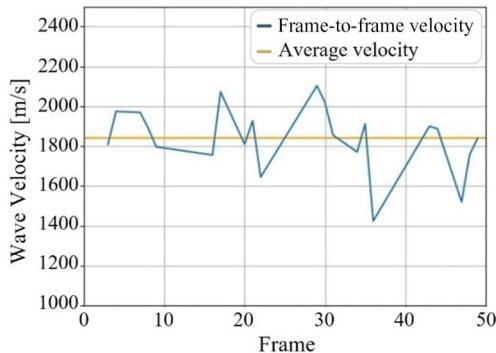
For annotation styles applied to traditional down-axis images, the linear distance between centroids from frame to frame can be related to the circumferential distance traveled. The wave velocity is calculated using Eqs. (3–5), which were outlined in the previous section. As mentioned in the previous subsection, annotations applied to the lin correspond directly to wave front angular locations. Therefore, these velocity calculations may be performed using only Eq. (5), where  $\theta_i$  is the 1-D shift of the lin annotation between two consecutive frames. Because these calculations are performed between individual images, substantial fluctuations are expected. The average velocity can also be found as a steady-state measurement. Using the S3 annotation style, velocity calculations were performed on a series of 50 images and are shown in Fig. 18. Larger values like that experienced at frame 43 are associated with images where wave

visibility is partially obstructed as an artifact of the experimental setup. With the exception of this outlier, the calculated velocities from frame to frame show more consistent values. To again highlight the benefits of the square annotation style as compared to the rectangular, the velocity across the same set of images is calculated using the S3 and R2 annotation sets, respectively, as shown in Figs. 18 and 19. Although the velocity data shown in Fig. 18 may not appear to offer drastic improvements over that in Fig. 19, the lower variance is beneficial to a more accurate average velocity calculation as well as frame-to-frame velocities. Reduction in the velocity variance associated with differing annotation styles does follow the trend suggested by the previously calculated Taubin RMSE. It is important to point out that although wave velocity calculations are not novel in the RDE community, this method attempts to perform these calculations with a time-step resolution of 20  $\mu\text{s}$  or a 50 kHz frequency. Additionally, a reliable measure of average velocity can be found with as few as 10 frames, or a 200  $\mu\text{s}$  window of operation.

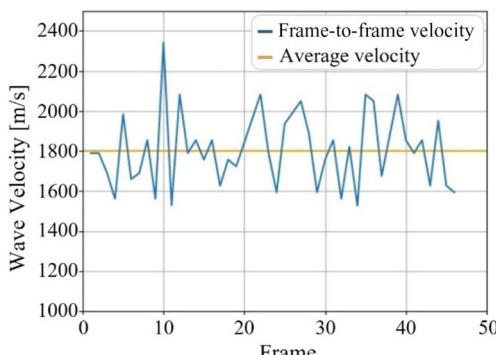
Eliminating the influence of the Taubin RMSE, velocity calculations using the linearized annotation style are shown in Fig. 20. Although the oscillation of the calculated velocities may seem more



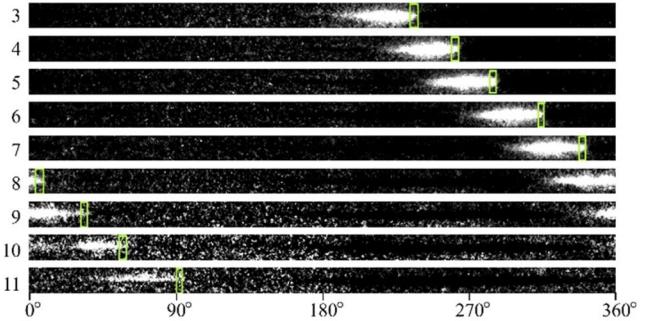
**Fig. 18** Wave velocity calculated across 50 frames using annotation S3.



**Fig. 19** Wave velocity calculated across 50 frames using R2.



**Fig. 20** Wave velocity calculated across 50 frames using linearized network.



**Fig. 21** Linearized image annotation predictions for images 3–11.

extreme in this case, it can be attributed to a higher number of predicted annotations across the same number of frames as well as an apparent detection of physical wave behavior. Figure 21 displays the annotations predicted for frames 3–11 using the linearized annotation style. Visualizing lines fit to the annotation locations across the time and unwrapped circumferential axes is reminiscent of detonation surface plots [6]. Reiterating the benefit of object detection proposed in this study, the annotations of each wave front offer a frame-to-frame velocity calculation that is more computationally efficient and requires less frames than the traditional detonation surface plotting method. The detonation wave propagating through Fig. 21 appears to change proportions by elongating, thereby affecting the wave front locations, and therefore introducing the velocity oscillations. The largest velocity peak shown in Fig. 20 is at frame 10, corresponding to the velocity calculated between frames 10 and 11. From frames 9 to 10, the wave appears to shorten, reducing the velocity; and then from frames 10 to 11, the wave elongates and drastically increases the circumferential path traveled. This behavior could be an artifact of imaging or a detection of the short timescale galloping as a local response to the fill region or other phenomena. Further work to identify the causation of this frame-to-frame behavior is necessary. Nevertheless, Fig. 21 shows that the velocity oscillations plotted in Fig. 20 can be attributed to the physical wave behavior captured in the high-speed images.

With the trained network, object detection is performed at approximately 1.5 frames per second for  $301 \times 301$  images and 9.5 frames per second for the linearized  $32 \times 736$  images using only the computer's Central Processing Unit (CPU). Undoubtedly, this speed could be drastically improved by using a Graphics Processing Unit (GPU) if detection speeds were of enough importance to the given study. It should be noted that real-time operation does not require that the wave speed be processed for every frame (at 50,000 fps) but rather at a speed sufficient to monitor wave velocity during steady operation. Compared to other object detection networks, the YOLO family relies on a single CNN, making high detection speeds possible. Based on the results outlined here and the experience related to other previous neural network success, it is believed that this method has produced a useful tool suitable for expedited postprocessing, and it serves to give further insight into the benefits of machine learning within the PGC community.

### E. Uncertainty Analysis

Experimental uncertainty for the wave velocity was estimated based on the method outlined by Moffat [33] by introducing a bias uncertainty for each of the measured quantities and calculating a precision uncertainty based on test statistics (frame-to-frame velocity calculations). The bias uncertainties were determined based on manufacturer specifications and engineering assumptions, and they were perturbed sequentially. The values that were perturbed in Eqs. (3–5) for the S3 and R2 annotation boxes were the annotation box centroid locations ( $x$  and  $y \pm 1$  pixels), the Taubin radius ( $r_{\text{Taub}} \pm 1$  pixel), the camera frame rate (FR  $\pm 2\%$ ), and the RDE radius ( $r_{\text{RDE}} \pm 0.10$  mm). The  $x$  and  $y$  pixel uncertainties were assumed to be approximately 5% of the width of the annotation boxes, whereas the Taubin radius uncertainty was determined from the RMSE. For the linearized annotation style relying only on Eq. (5), the values perturbed were the angle of rotation

$(\theta_i \pm 0.00856 \text{ rad})$ , the camera frame rate ( $\text{FR} \pm 2\%$ ), and the RDE radius ( $r_{\text{RDE}} \pm 0.10 \text{ mm}$ ). In the linearized format, only the angular bin was found from annotation centroid  $x$  locations, where each pixel width was equal to  $0.00856 \text{ rad}$  and each annotation was 10 pixels wide. Remaining consistent, pixel uncertainties were assumed to be approximately 5% of the width of the annotation boxes. Because 5% of the 10-pixel annotation width was less than 1 pixel, perturbations used  $\pm 1$  pixel or, as noted earlier,  $\pm 0.00856 \text{ rad}$ . This reduction of variables perturbed through a lesser number of equations will directly reduce the bias uncertainty. The camera frame rate uncertainty was estimated based on data reported by Robbe et al. [34], and the RDE radius accuracy was assumed based on machining tolerances and by considering that the centroid does not necessarily follow the centerline of the RDE annulus. Once all values had been perturbed, the positive perturbation and negative perturbation of each measurement were taken, and their absolute values were averaged. These values were then used to find the root-mean-square value of the bias uncertainty, which was assumed to be the same for the two annotation styles.

The second component is the precision uncertainty and is based on a Student's  $t$  distribution based on the number of tests performed and the test points collected (30 frames). Finally, the total uncertainty  $V_{\text{uncertainty}}$  can be determined from the bias uncertainty  $V_{\text{bias}}$  and the precision uncertainty  $V_{\text{precision}}$ , shown in Eq. (8) for the wave velocity:

$$V_{\text{uncertainty}} = \sqrt{V_{\text{bias}}^2 + V_{\text{precision}}^2} \quad (8)$$

The bias uncertainty was determined to be  $\pm 3.6\%$  for annotation styles S3 and R2, and it was  $\pm 2.7\%$  for the lin annotation style based on the measurement specifications discussed earlier in this paper. The precision uncertainties for the S3, R2, and lin annotation boxes, shown in Figs. 18–20, were  $\pm 2.4$ ,  $\pm 2.8$ , and  $\pm 2.7\%$ , respectively. The total uncertainty was then determined to be  $\pm 4.3\%$  for the S3 annotation boxes,  $\pm 4.6\%$  for the R2 annotation boxes, and  $\pm 3.8\%$  for the lin annotation style. The strength of the linear adaptation offers total uncertainty reductions of 0.5 and 0.8% as compared to its S3 and R2 competitors, respectively.

#### F. Network Tolerance and Benchmark

As previously mentioned, compared to image classification one of the expected benefits of object detection applied to down-axis RDE images is the ability to classify wave modes which are not directly represented in the training image set. Additionally, it was stated that YOLO learns representations of objects that are more generalizable as a result of contextual consideration of the entire image. Compared to other object detection methods, this is hoped to increase the network tolerance to new modes as well as varied down-axis RDE imaging arrangements in external laboratories. To evaluate the network on these metrics of tolerance, new testing data must be evaluated. To independently benchmark these aspects of network tolerance, the trained RDE YOLO network is tested on external images, a new internal camera arrangement, and new wave modes.

High-speed images of detonation waves within the experimental RDE at the Technical University of Berlin (TU Berlin) were provided by Bohon et al. [7,35,36]. The RDE at the TU Berlin is an 82.4 mm nominal diameter RDE, with images captured from the reflection of a warped down-axis mirror at a frame rate of 87,500 fps. In this comparison, the RDE size, the image acquisition components, and the frame rate differ from the current experimental setup. The external images from the TU Berlin are linearized to match image dimensions compatible with the trained YOL lin network. As shown in Fig. 22, the representation of the TU Berlin detonation wave does visibly differ from the National Energy Technology Laboratory (NETL) images in pixel variation and wave profile length. A correction for elliptical distortion in the external images [37] also introduces a small frame-to-frame deviation in radial location of the wave front, which is depicted as the vertical direction within linearized images. Still yet, the YOLO network provides promising wave detections for external images, which are shown in Fig. 22. Although detection sparsity is apparent, the absence of incorrect annotations is promis-

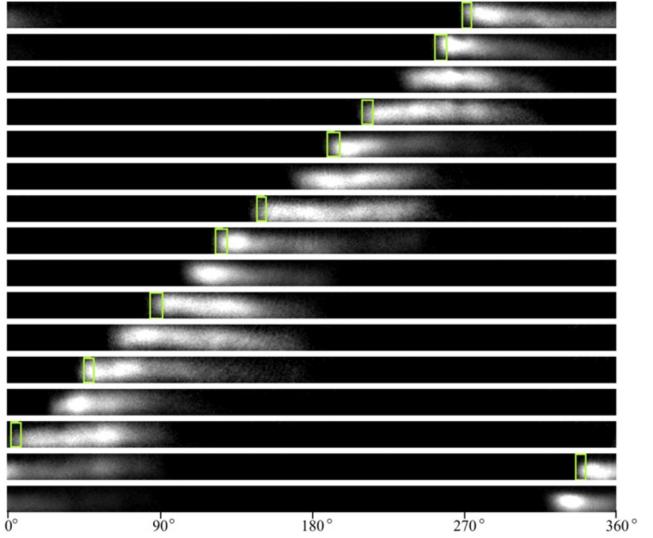


Fig. 22 Linearized image annotation predictions for external images provided by TU Berlin.

ing. Velocity calculated across these images using the YOLO lin and the detonation surface are 1367.3 and 1332.4 m/s, respectively. A portion of the discrepancy in calculated velocities may be attributed to circumferential differences in the corrected annulus geometry or the lower resolution of the detonation surface velocity output associated with the FFT across only 100 images. Nonetheless, the YOLO lin calculated a comparable velocity for external data using half as many images while also offering valuable insights on the wave location at a highly resolved time step of  $11.43 \mu\text{s}$ .

Within the NETL RDE facility, the high-speed imaging arrangement is modified for increased image quality. Specifically, the high-speed camera and intensifier are moved to an axis-aligned orientation, negating the use of a mirror required by the perpendicular arrangement used throughout the training set. Images in the axis-aligned setup are also recorded at 60,000 fps, resulting in shortened exposure times as compared to the training set. The representation of individual waves is shortened due to shorter exposure times, which can be seen among waves in Fig. 23 as compared to waves depicted in Fig. 21. In addition to an altered camera alignment, the images shown in Figs. 23 and 24 correspond to an injector geometry and pixel resolution differing from that captured by images in the training set. The YOLO annotations of a five-wave mode are shown in Fig. 23. This output not only demonstrates the network's ability to process images from a varied experimental setup but also a tolerance to new wave modes not present in the training set. Wave modes in the training set include an overall maximum of four waves (being two sets of counter-rotating waves) and a maximum of three waves for corotational modes. In each linearized image within Fig. 23, all five wave fronts are properly detected, with the exception of one wave in frame 6, for which the trail is mistaken for a wave front.

Considering an additional aspect of wave mode tolerance, beyond new quantities of wave counts, the wave-to-wave interactions

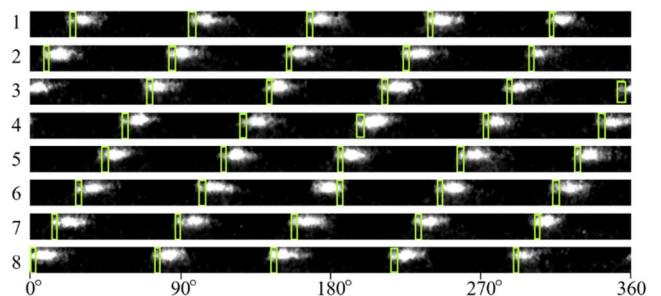
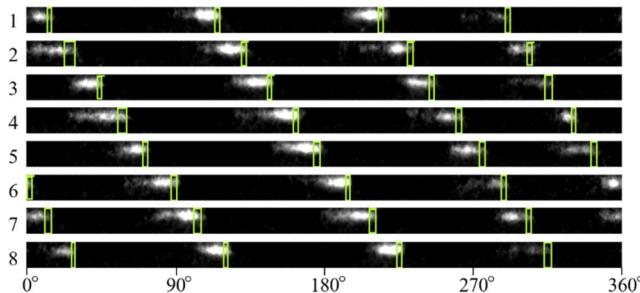


Fig. 23 Linearized image annotation predictions for five-wave mode with axis-aligned imaging.



**Fig. 24** Linearized image annotation predictions for four-wave galloping mode with axis-aligned imaging.

observed as galloping waves [38] are not present in the training set. Although the cause of galloping waves is not yet fully known, the galloping wave mode is present and stable at repeatable operating conditions. Within the galloping mode, the wave-to-wave pixel intensity varies as the local wave spacing oscillates. The YOLO lin detections for a four-wave galloping mode are shown in Fig. 24. Although there are no four-wave corotational wave modes, or any galloping modes in the training set, the trained network is able to accurately detect wave fronts in Fig. 24. Waves with varied representation throughout Fig. 24, including momentarily “weak” waves such as the detection shown in frame 1 at approximately 300 deg, are well generalized by the trained network.

The results shown in Figs. 22–24 demonstrate that the YOLO network is not observably overfit to the original set of wave modes or image configuration, including the facility, RDE geometry, camera configuration, and frame rate. Table 1 benchmarks the trained YOLO network against the highly robust, widely used detonation surface method. Considering the varying RDE performance as well as new mode tolerance, both of which are metrics satisfied by the detonation surface, the results presented in the current subsection show that YOLO is also capable. Specifically, properly annotated images from the TU Berlin and NETL collectively represent YOLO’s tolerance to varied RDE size, camera alignment, frame rate, injector schemes, and pixel resolution. Images depicting five corotational waves and four galloping corotational waves illustrate YOLO’s ability to classify wave modes not present in the training set.

It is uncommon to report the data processing speeds of detonation surfaces. For the purpose of comparison, the detonation surface method is timed for 100 images, resulting in a classification speed of 0.5892 seconds per image. The data processing time, documented in Table 2, indicates the time to generate a single velocity measurement. For the detonation surface, individual velocity readings require analysis of at least 100 sequential images, resulting in a data processing time of 58.92 s, and a diagnostic feedback rate of 0.017 Hz. The YOLO lin provides annotations at a rate of 0.105 seconds per image, corresponding to the inference time. However, to generate a velocity output, two sequential images must be analyzed. This corresponds to a data processing time of 0.210 s. The YOLO lin has a diagnostic feedback rate of 9.5 Hz because, after an initial velocity calculation, the annotations from only one additional frame are used for subsequent velocity calculations. Of particular interest to the current study, the diagnostic time-step resolution is documented in Table 2, which indicates the time over which an individual velocity reading is computed. Both values assume a frame rate of 50 kfps. As the frame rate increases, the diagnostic time-step resolution is expected to improve for the YOLO lin but will likely remain unchanged for the detonation surface. This is because the detonation surface average velocities rely on the frequency

analysis of multiple wave revolutions around the annulus, whereas the YOLO lin is computing frame-to-frame velocities. So, although an increased frame rate may improve the resolution of the detonation surface, the number of frames required to perform an average velocity calculation will also increase. At a 50 kfps frame rate, the detonation surface and the YOLO lin achieve diagnostic time-step resolutions of 2 and 0.02 ms, or 20  $\mu$ s, respectively. An increased frame rate of 60 kfps at NETL results in an improved time-step resolution of 17  $\mu$ s; and the external images captured at TU Berlin at a frame rate of 87.5 kfps offer a further refined time-step resolution of 11  $\mu$ s. Data processing times are determined using CPUs on a laboratory Dell Precision 7550 laptop with an Intel® Core™ i9-10885H CPU at 2.40 GHz with 64.0 GB of RAM.

It should be noted that the detonation surface is not optimized for increased processing speeds; although the YOLO lin outperforms in short timescales, the detonation surface method retains many strengths as a robust image analysis technique. The demonstrated performance of the YOLO lin offers more finely resolved time steps for studies interested in short timescale phenomena, such as galloping or counter-rotating interactions, with data processing speeds potentially offering a route toward real-time object detection (offering wave mode classification alongside velocities) in the RDE environment.

## V. Conclusions

The detection and tracking of detonation waves present in a RDE annulus during combustor operation is critical to advance the monitoring of RDEs as they progress toward gas turbine engine integration. A methodology for wave detection in down-axis images using a computer vision YOLO network was demonstrated in this work through analysis of high-speed camera images. The goal of this work was to use the YOLO architecture to report the location of each wave within the image as well as the wave direction for multiwave modal behavior in an experimental RDE. To do so, the YOLO network was first trained and validated using a chosen set of manually annotated and treated images. Separate efforts were outlined to alter the IOU and confidence thresholds to improve the annotation outputs according to the physical understanding of the RDE. The trained network was equipped to predict annotation boxes around each detonation wave, and the centroid of each annotation could be tracked through a series of images. The progression of annotation centroids was used to calculate wave velocity from frame to frame as well as through a series of images to determine an average wave velocity. The sizes and proportions of the annotation styles were found to have a discernable impact on the reported centroids and their radial errors with respect to a Taubin fit circle. The increased radius variation associated with larger, rectangular annotations resulted in higher uncertainty for the velocity calculations. The small, square annotations, however, experienced a higher classification error due to a reduction in the wave features present within the annotation area. An additional network was trained to analyze linearized (or unwrapped) images containing only the RDE annulus pixels. The lin offered a higher volume of annotation predictions and a significantly higher processing rate due to the more concise pixel arrangement, and it reduced velocity uncertainty by reducing dependence on the annotation coordinates to only angular locations. Therefore, the resultant recommendation is that studies more interested in wave direction labeling use larger, rectangular annotations; whereas studies interested more in wave velocity or processing speeds should use linearized images with corresponding binned annotations. Because the goal of this study is wave speed calculations, the latter annotation style was selected and resulted in an overall uncertainty of  $\pm 3.8\%$ .

This work builds on the previous work and experience of training a neural network for analysis of high-speed down-axis RDE images [2]. Unlike the previous study, the current method does not rely on pressure data to determine wave velocities and is compatible with wave modes that do not exist in the training image set. Both studies worked to find wave velocity through image processing, which is an effort that was previously explored within the RDE/PGC community; and both studies introduced computer vision to accelerate the

**Table 2** Comparison of conventional and YOLO techniques

| Method             | Data processing time, s | Diagnostic feedback rate, Hz | Diagnostic time-step resolution, ms | Varying RDE performance | New mode tolerance |
|--------------------|-------------------------|------------------------------|-------------------------------------|-------------------------|--------------------|
| Detonation surface | 58.92                   | 0.017*                       | 2                                   | ✓                       | ✓                  |
| YOLO               | 0.210                   | 9.50*                        | 0.02                                | ✓                       | ✓                  |

\* indicates results obtained in post-processing.

process. The current method calculates the frame-to-frame velocity with a time-step resolution of 20  $\mu\text{s}$  or a 50 kHz frequency, and it can process and annotate images at 1.5 frames per second for  $301 \times 301$  images and 9.5 frames per second for the linearized  $32 \times 736$  images. The RDE YOLO network offers frame-to-frame velocities across short windows of images at processing times that outpace conventional postprocessing techniques, which determine only an average velocity typically over 1000 frames. As computer vision and machine vision evolve alongside RDE technology, improved processing time and refined networks will allow for the progressive development of more efficient monitoring for industrial RDEs. The benefits of object detection are expected to extend to wave strength and parasitic combustion quantification in future studies. This demonstrated approach offers an early example of neural network application to experimental RDE research.

### Acknowledgments

This project was funded by the U.S. Department of Energy (DOE) National Energy Technology Laboratory (NETL), in part, through a site support contract. This research was supported in part by an appointment to the U.S. DOE Postgraduate Research Program at the NETL and administered by the Oak Ridge Institute for Science and Education. This work was performed in support of the U.S. Department of Energy's Fossil Energy and Carbon Management's Advanced Turbines Program and executed through the National Energy Technology Laboratory Research and Innovation Center. The research was greatly assisted by additional images provided by Carson Slabaugh at Purdue University.

Neither the U.S. Government nor any agency thereof, nor any of their employees, nor the support contractor, nor any of their employees make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the U.S. Government or any agency thereof. The views and opinions of the authors expressed herein do not necessarily state or reflect those of the U.S. Government or any agency thereof.

### References

- [1] Nordeen, C., Schwer, D., Schauer, F., Hoke, J., Cetegen, B., and Barber, T., "Thermodynamic Modeling of a Rotating Detonation Engine," *49th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, AIAA Paper 2011-0803, Jan. 2011. <https://doi.org/10.2514/6.2011-803>
- [2] Johnson, K. B., Ferguson, D. H., Tempke, R. S., and Nix, A. C., "Application of a Convolutional Neural Network for Wave Mode Identification in a Rotating Detonation Combustor Using High-Speed Imaging," *Journal of Thermal Science and Engineering Applications*, Vol. 13, No. 6, 2021, Paper 061021. <https://doi.org/10.1115/1.4049868>
- [3] Chacon, F., Feleco, A. D., and Gamba, M., "Secondary Waves Dynamics and Their Impact on Detonation Structure in Rotating Detonation Combustors," *Shock Waves*, Vol. 31, No. 7, 2021, pp. 675–702. <https://doi.org/10.1007/s00193-021-01034-6>
- [4] Hsu, P. S., Slipchenko, M. N., Jiang, N., Fugger, C. A., Webb, A. M., Athmanathan, V., Meyer, T. R., and Roy, S., "Megahertz-Rate OH Planar Laser-Induced Fluorescence Imaging in a Rotating Detonation Combustor," *Optics Letters*, Vol. 45, No. 20, 2020, pp. 5776–5779. <https://doi.org/10.1364/OL.403199>
- [5] Johnson, K., Ferguson, D. H., and Nix, A., "Validation of Cross-Correlation Detonation Wave Mode Identification Through High-Speed Image Analysis," *AIAA SciTech 2020 Forum*, AIAA Paper 2020-1179, Jan. 2020. <https://doi.org/10.2514/6.2020-1179>
- [6] Bennewitz, J. W., Bigler, B. R., Schumaker, S. A., and Hargus, W. A., "Automated Image Processing Method to Quantify Rotating Detonation Wave Behavior," *Review of Scientific Instruments*, Vol. 90, No. 6, 2019, Paper 065106. <https://doi.org/10.1063/1.5067256>
- [7] Bohon, M. D., Bluemner, R., Paschereit, C. O., and Gutmark, E. J., "High-Speed Imaging of Wave Modes in an RDC," *Experimental Thermal and Fluid Science*, Vol. 102, April 2019, pp. 28–37. <https://doi.org/10.1016/j.expthermflusci.2018.10.031>
- [8] Ballard, D. H., and Brown, C., *Computer Vision*, 1st ed., Prentice-Hall, Englewood Cliffs, NJ, 1982, p. xiii.
- [9] Frazzoli, E., Munther, A. D., and Feron, E., "Real-Time Motion Planning for Agile Autonomous Vehicles," *Journal of Guidance, Control, and Dynamics*, Vol. 25, No. 1, 2002, pp. 116–129. <https://doi.org/10.2514/2.4856>
- [10] Reji, R., Sojan Lal, P., Philip, A. M., and Vishnu, V., "A Compact Deep Learning Model for Robust Facial Expression Recognition," *International Journal of Engineering and Advanced Technology*, Vol. 8, No. 6, 2019, pp. 2956–2960. <https://doi.org/10.35940/ijeat.F8724.088619>
- [11] Huang, K. T., "Integrating Computer Vision and Non-Linear Optimization for Automated Deformable Registration of 3D Medical Images," *Physics in Medicine and Biology*, Vol. 64, No. 13, 2019, Paper 135014. <https://doi.org/10.1088/1361-6560/ab2202>
- [12] Sentieiro, J., "A Computer Vision System for the Characterization and Classification of Flames in Glass Furnaces," *IEEE Transactions on Industry Applications*, Vol. 29, No. 3, 1996, pp. 470–478. <https://doi.org/10.1109/28.222414>
- [13] Zhou, L., Song, Y., Ji, W., and Wei, H., "Machine Learning For Combustion," *Energy and AI*, Vol. 7, Jan. 2022, Paper 100128. <https://doi.org/10.1016/j.egyai.2021.100128>
- [14] Barwey, S., Prakash, S., Hassanaly, M., and Raman, V., "Data-Driven Classification and Modeling of Combustion Regimes in Detonation Waves," *Flow, Turbulence and Combustion*, Vol. 106, No. 4, 2021, pp. 1065–1089. <https://doi.org/10.1007/s10494-020-00176-4>
- [15] D'Alessio, G., Attili, A., Cuoci, A., Pitsch, H., and Parente, A., "Analysis of Turbulent Reacting Jets via Principal Component Analysis," *Data Analysis for Direct Numerical Simulations of Turbulent Combustion*, Springer, Cham, Switzerland, 2020, pp. 233–251.
- [16] Chen, Y., and Wang, J., "Computer Vision-Based Approach for Reading Analog Multimeter," *Applied Sciences*, Vol. 8, No. 8, 2018, pp. 1268–1268. <https://doi.org/10.3390/app8081268>
- [17] Boillet, M., Kermorvant, C., and Paquet, T., "Robust Text Line Detection in Historical Documents: Learning and Evaluation Methods," *International Journal on Document Analysis and Recognition*, Vol. 25, No. 2, 2022, pp. 95–114. <https://doi.org/10.1007/s10032-022-00395-7>
- [18] Kim, W., Cho, H., Kim, J., Kim, B., and Lee, S., "Yolo-Based Simultaneous Target Detection and Classification in Automotive FMCW Radar Systems," *Sensors*, Vol. 20, No. 10, 2020, Paper 2897. <https://doi.org/10.3390/s20102897>
- [19] Ko, K. E., Yang, S.-H., and Jang, I., "Real-time Tomato Ripeness Classification System Based on Deep Learning Model for Object Detection," *Journal of Institute of Control, Robotics and Systems*, Vol. 24, No. 11, 2018, pp. 999–1004. <https://doi.org/10.5302/J.ICRDS.2018.18.0166>
- [20] Virasova, A. Y., Klimov, D. I., Khromov, O. E., Gubaidullin, I. R., and Oreshko, V. V., "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," *Radioengineering*, Vol. 85, No. 9, 2021, pp. 115–126. <https://doi.org/10.18127/j00338486-202109-11>
- [21] Girshick, R., "Fast R-CNN," *2015 IEEE International Conference on Computer Vision (ICCV)*, IEEE Publ., Piscataway, NJ, 2015, pp. 1440–1448. <https://doi.org/10.1109/ICCV.2015.169>
- [22] Ren, S., He, K., Girshick, R., and Sun, J., "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 39, No. 6, 2017, pp. 1137–1149. <https://doi.org/10.1109/TPAMI.2016.2577031>
- [23] Redmon, J., Divvala, S., Farhadi, A., and Girshick, R., "You Only Look Once: Unified, Real-Time Object Detection," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Publ., Piscataway, NJ, 2016, pp. 779–788. <https://doi.org/10.1109/CVPR.2016.91>
- [24] Journell, C. L., Gejji, R. M., Walters, I. V., Lemcherfi, A. I., Slabaugh, C. D., and Stout, J. B., "High-Speed Diagnostics in a Natural Gas-Air Rotating Detonation Engine," *Journal of Propulsion and Power*, Vol. 36, No. 4, 2020, pp. 498–507. <https://doi.org/10.2514/1.B37740>
- [25] Walters, I. V., Journell, C. L., Lemcherfi, A., Gejji, R. M., Heister, S. D., and Slabaugh, C. D., "Operability of a Natural Gas-Air Rotating Detonation Engine," *Journal of Propulsion and Power*, Vol. 36,

- No. 3, 2020, pp. 453–464.  
<https://doi.org/10.2514/1.B37735>
- [26] Walters, I. V., Lemcherfi, A., Gejji, R. M., Heister, S. D., and Slabaugh, C. D., “Performance Characterization of a Natural Gas-Air Rotating Detonation Engine,” *Journal of Propulsion and Power*, Vol. 37, No. 2, 2021, pp. 292–304.  
<https://doi.org/10.2514/1.B38087>
- [27] Taubin, G., “Estimation of Planar Curves, Surfaces, and Nonplanar Space Curves Defined by Implicit Equations with Applications to Edge and Range Image Segmentation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13, No. 11, 1991, pp. 1115–1138.  
<https://doi.org/10.1109/34.103273>
- [28] Microsoft, “Visual Object Tagging Tool (2.2.0) [Software],” Ver. 2.2.0, GitHub, Public archive, 2020, <https://github.com/Microsoft/VoTT/releases> [retrieved 10 Nov. 2020].
- [29] Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A., “The Pascal Visual Object Classes (VOC) Challenge,” *International Journal of Computer Vision*, Vol. 88, No. 2, 2010, pp. 303–338.  
<https://doi.org/10.1007/s11263-009-0275-4>
- [30] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L., “ImageNet Large Scale Visual Recognition Challenge,” *International Journal of Computer Vision*, Vol. 115, No. 3, 2015, pp. 211–252.  
<https://doi.org/10.1007/s11263-015-0816-y>
- [31] Yuan, D., Shu, X., Fan, N., Chang, X., Liu, Q., and He, Z., “Accurate Bounding-Box Regression with Distance-IoU Loss for Visual Tracking,” *Journal of Visual Communication and Image Representation*, Vol. 83, Feb. 2022, Paper 103428.  
<https://doi.org/10.1016/j.jvcir.2021.103428>
- [32] Wu, G., and Li, Y., “Non-Maximum Suppression for Object Detection Based on the Chaotic Whale Optimization Algorithm,” *Journal of Visual Communication and Image Representation*, Vol. 74, Jan. 2021,  
Paper 102985.  
<https://doi.org/10.1016/j.jvcir.2020.102985>
- [33] Moffat, R. J., “Describing the Uncertainties in Experimental Results,” *Experimental Thermal and Fluid Science*, Vol. 1, No. 1, 1988, pp. 3–17.  
[https://doi.org/10.1016/0894-1777\(88\)90043-X](https://doi.org/10.1016/0894-1777(88)90043-X)
- [34] Robbe, C., Nsiampa, N., Ouakara, A., and Papy, A., “Quantification of the Uncertainties of High-Speed Camera Measurements,” *International Journal of Metrology and Quality Engineering*, Vol. 5, No. 2, 2014, Paper 201.  
<https://doi.org/10.1051/ijmqe/2014007>
- [35] Bohon, M. D., Bluemner, R., Paschereit, C. O., and Gutmark, E. J., “Cross-Correlation as a Tool for Measuring RDC Wave Speed, Direction, and Complexity,” *2018 Joint Propulsion Conference*, AIAA Paper 2018-4569, 2018.  
<https://doi.org/10.2514/6.2018-4569>
- [36] Bohon, M. D., Orchini, A., Bluemner, R., Paschereit, C. O., and Gutmark, E. J., “Dynamic Mode Decomposition Analysis of Rotating Detonation Waves,” *Shock Waves*, Vol. 31, No. 7, 2021, pp. 637–649.  
<https://doi.org/10.1007/s00193-020-00975-8>
- [37] Johnson, K., Ferguson, D. H., and Nix, A. C., “Survey of Machine Learning Based RDE Diagnostics: Evaluation and Comparison for Broad Application in Experimental Facilities,” *AIAA SciTech 2023 Forum*, AIAA Paper 2023-2394, Jan. 2023.
- [38] Johnson, K., Weber, J., Ferguson, D. H., and Nix, A. C., “Analysis of Quasi-Steady, Transitional, and Short Timescale Galloping within Rotating Detonation Engines,” *AIAA SciTech 2023 Forum*, AIAA Paper 2023-0931, Jan. 2023.

V. Raman  
Associate Editor