

Research Paper

Measurement of hybrid rocket solid fuel regression rate for a slab burner using deep learning

Gabriel Surina III^a, Georgios Georgalis^b, Siddhant S. Aphale^a, Abani Patra^b, Paul E. DesJardin^{a,*}

^a Department of Mechanical and Aerospace Engineering, University at Buffalo, The State University of New York, Buffalo, NY 14260-4400, USA

^b Data Intensive Studies Center, Tufts University, Medford, MA 02155, USA



ARTICLE INFO

Keywords:

Hybrid fuel
Regression rate measurement
Convolutional neural network
Monte-Carlo dropout
Model uncertainty

ABSTRACT

This study presents an imaging-based deep learning tool to measure the fuel regression rate in a 2D slab burner experiment for hybrid rocket fuels. The slab burner experiment is designed to verify mechanistic models of reacting boundary layer combustion in hybrid rockets by the measurement of fuel regression rates. A DSLR camera with a high intensity flash is used to capture images throughout the burn and the images are then used to find the fuel boundary to calculate the regression rate. A U-net convolutional neural network architecture is explored to segment the fuel from the experimental images. A Monte-Carlo Dropout process is used to quantify the regression rate uncertainty produced from the network. The U-net computed regression rates are compared with values from other techniques from literature and show error less than 10%. An oxidizer flux dependency study is performed and shows the U-net predictions of regression rates are accurate and independent of the oxidizer flux, when the images in the training set are not over-saturated. Training with monochrome images is explored and is not successful at predicting the fuel regression rate from images with high noise. The network is superior at filtering out noise introduced by soot, pitting, and wax deposition on the chamber glass as well as the flame when compared to traditional image processing techniques, such as threshold binary conversion and spatial filtering. U-net consistently provides low error image segmentations to allow accurate computation of the regression rate of the fuel.

1. Introduction

Hybrid rockets are recently gaining attention and becoming a competitive propulsion system because they have the benefits of solid and liquid rockets in one platform [1,2]. The fuel and oxidizer are in two different states: typically the fuel is a solid and the oxidizer is a gas/liquid, providing a highly dense fuel source similar to a bi-propellant solid motor but with the throttle control of a liquid motor [3–5]. The fuel regression rate ($\dot{r}$) corresponds to the rate that the fuel recedes over the course of a burn and is an important measurement since it directly defines the motor thrust and geometry (e.g., a higher regression rate results in smaller combustion chambers) [6]. Low fuel regression rates correspond to low fuel mass flow rates, leading to excess oxidizer passing through the motor, which reduces the available thrust and therefore reduces the overall efficiency [7]. From a prediction perspective, the regression rate is a simple measurement that can be used to verify model estimates.

The simplest method to measure temporally and spatially averaged regression rate is to measure the fuel grain mass before and after the burn, then divide by the time of the burn, as performed in [8–10].

Measuring the regression rate in this way may be subject to error caused by part of the fuel mass not combusting and melting away (e.g., for solid fuels that have a low melting point such as paraffin wax). Another drawback to weighing the fuel grain is that it does not provide data of local and instantaneous regression rates which is useful for validating regression rate models [6]. Other methods of regression rate measurement include using combustion chamber pressure [11], x-ray radiography [3], initial and final port diameter [6,12,13], and ultrasonic transducers [14–16]. More recently, [5] developed an imaging-based regression rate measurement for a 2D slab burner experiment. They collected images of the fuel profiles captured continuously throughout the burn and manually traced the fuel boundary to segment the fuel from the background (binary masks where a value of “0” corresponds to background and a value of “1” corresponds to fuel). The regression rate was then computed by measuring the fuel height variation over the burn time.

Even if manually segmenting the fuel and noise from experimental images is accurate, it is certainly not the most efficient and repeatable process for a large number of images and experimental dataset. The

* Corresponding author.

E-mail address: ped3@buffalo.edu (P.E. DesJardin).

<https://doi.org/10.1016/j.actaastro.2021.09.046>

Received 18 August 2021; Received in revised form 15 September 2021; Accepted 30 September 2021

Available online 9 October 2021

0094-5765/© 2021 IAA. Published by Elsevier Ltd. All rights reserved.

task of solid fuel image segmentation for measuring the fuel regression rate can benefit from advances in deep learning processes. Image segmentation refers to the process of classifying objects and identifying their location in an image. Modern image processing tools based on deep learning, such as convolutional neural networks (CNNs) have been used to segment images for autonomous driving [17] and medical applications [18]. CNNs are primarily made up of convolutional layers, intermediate activation function (or transfer function) layers, and pooling layers. Convolutional layers apply kernels (or filters) of weights that convolve with the image creating a feature map to extract important learning features from the images. The activation function layers apply a non-linear function, a Rectified Linear Unit (ReLU), that determines the output of previous convolution nodes in the neural network and whether they will be activated. The pooling layers reduce the size of the feature map and also help retain the useful information for the segmentation task [19]. The pooled feature maps result in smaller number of parameters in the network and reduced computational expense. Training of a CNN refers to the process of computing the values of the kernel weights. After initialization, the weights are updated using the gradient descent optimization method after each pass through the training data (or epoch) [20].

U-net is a widely used CNN architecture that was initially created for biomedical image segmentation with a low number of training data [21]. Examples of U-net based architecture applications include infant brain segmentation [22], nuclei segmentation for cancer grading and cell detection/classification [23], chronic stroke lesion segmentation [24], liver and tumor segmentation [25], retinal vessel segmentation for discover and treatment of ocular diseases [26], and even cloud and shadow segmentation [27]. Although, U-net has mostly been applied in the medical field, more recent studies have demonstrated applications specifically in combustion, such as aluminum droplet combustion detection in solid propellant from shadowgraph images [28], flame segmentation to locate the hottest region for fire suppression [29], and segment coolant film and hot exhaust gases in a rocket combustion chamber [30]. Neural networks in general have also been recently utilized in the combustion field, including maximizing power density in proton exchange membrane fuel cells [31], creating a fuel consumption model to predict the most efficient route for truck transportation [32], and to classify solid fuels based on ash content, volatile matter, and fixed carbon [33]. Similar machine learning processes have been used in hybrid rockets to recognize the different burning phases of solid fuel in a similar 2D slab burner experiment [34]. The most relevant work to the goal of this study is the particle shape and size regression detection, an example, BubCNN, presented by [35], where neural networks were used to detect the shape and size of bubbles in a gas–liquid multiphase flow while segmenting them from the background. Another example would be [36] where the U-net was used to segment bubble images and classify particle shapes and sizes. The work from [35] and [36] however do not have noise from soot, pitting, wax, and flame interfering with the imaging.

The overall research focus of this study is on automating the manual regression rate measurement technique in [5] using U-net to produce the fuel segmentation masks instead of manually tracing them. The manual approach is used as the ground truth and serves as a baseline to validate the U-net results. The goal of this paper is to evaluate whether the U-net architecture is a viable and self-consistent way to produce segmented fuel masks to measure the fuel regression rate. To accomplish this goal, a series of U-net models of similar architecture is built. Each model is trained on a subset of the data collected in [5], a 2D slab burner experiment with four different oxidizer fluxes (5.91, 9.58, 18.59, 22.19 [kg/m²-s]). The data included, in total, 150 RGB images and their corresponding manually segmented fuel masks. The U-net models are then used to predict the segmented fuel masks from the input images together with estimations of model uncertainty. The fuel masks and uncertainty information are then used to measure the fuel regression rate and are compared with values reported from the manual

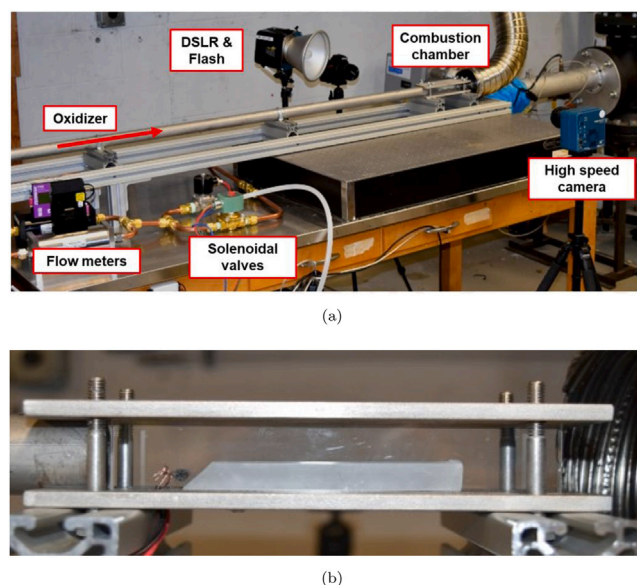


Fig. 1. The 2D slab burner (a) overall experimental setup, and (b) side view of the combustion chamber showing paraffin wax fuel [5].

process. Lastly, the network robustness and limitations are investigated through an oxidizer flux independency and monochromatic data study.

The rest of this paper is organized as follows. Section 2 presents the experimental setup. Section 3 describes various methods to produce the fuel segmentation masks including the U-net architecture. Results and associated discussion are presented in Section 4. Conclusions of this study are in Section 5.

2. Experimental setup

Fig. 1a shows the 2D slab burner experiment for paraffin wax-gaseous oxygen hybrid combustion. The experiments are conducted in the setup that is based on [5] and [8]. The experiment uses solid paraffin wax (or candle wax) as the fuel because of its low cost and high regression rate characteristics. The paraffin used is laboratory grade manufactured by the Carolina Biological Supply Company. The oxidizer is controlled using a solenoid valve and monitored using an Omega FMA 1744a mass flow meter. The oxidizer flows through a 2.54 cm inner diameter stainless steel pipe with a length of 1.83 m to provide a fully developed flow [8]. The oxidizer enters the combustion chamber shown in Fig. 1b where the fuel sample is placed between two 3 mm thick borosilicate glasses for optical access with stainless steel plates on top and bottom.

A CMOS based Nikon D5600 DSLR camera focused on the entire chamber is setup approximately 500 mm away, perpendicular to the chamber. A digital stopwatch is setup in the frame of the image to obtain image time to a millisecond accuracy. The fuel is ignited with ethylene vinyl acetate (EVA) and steel wool attached to a nichrome wire supplied with 12 V, as performed by [8]. The experimental time ranges from 5–10 s depending on the incoming oxidizer flux. Approximately 40 continuous images are taken for each case during this time. The amount of images are limited by the flash's ability to recharge.

The slab burner experiments are conducted at four different oxidizer fluxes (5.91, 9.58, 18.59, 22.19 [kg/m²-s]). As shown in Fig. 2, the wax profile is blocked by flame that brightens with increasing oxidizer flux. To enable viewing the fuel through the flame, a high intensity flash is used to saturate the flame and illuminate the fuel surface, however a flame ghost image remains. Fig. 3 shows a series of images captured with the high intensity flash. Although the flash allows for imaging of the regressing fuel, it contains noise. With increasing oxidizer flux, the



Fig. 2. An example of the side image of the wax without using a high intensity flash.

flame brightens and can exceed the intensity of flash, leading to flame traces in the images (Fig. 3b). The flash intensity can be increased but it results in over-saturation (Fig. 3c). Apart from these issues, the optical windows are also prone to noise introduced by molten wax splashing on the glasses and soot pitting (Fig. 3d). These issues, especially flame interference and wax on the glass make it difficult to automate the accurate tracing of the wax profile resulting error in regression rate estimates. Thus, a neural network enables automating the tracing of the fuel profile in the noisy images.

3. Methods of image processing

The regression rates are calculated by tracking the height of the wax over time in successive images (from the fuel masks), fitting a cubic polynomial curve, and finally taking its derivative, consistent with [5]. The binary images are processed by starting from the top of the image and stepping down each pixel vertically while searching for the profile height. The profile is constrained to only regress or

remain constant, i.e. the height of the wax can only decrease or remain constant. If the height increases in time, that data is neglected. Fig. 4 shows representative binary masks of the regressing fuel with the profile tracking in red lines with circles, the profile height (h) for each column of pixels is used in the regression rate estimation. The accurate segmented binary masks are essential to estimate accurate regression rates. Four image processing methods are investigated and compared for the accuracy in estimating the regression rates. The methods of the image processing techniques include using (i) threshold, (ii) Threshold Last Image Subtraction (TLIS), (iii) spatial filtering, and (iv) U-net. Thresholding represents the least complex image processing method and has little to no ability to filter out noise. TLIS is a method that removes all of the noise but removes significant amounts of data in doing so. The spatial filtering method removes all of the small clusters of noise but amplifies large noise clusters. The U-net model artificially chooses the best filters to apply, including image processing techniques that are not readily available, providing the best results, if trained well enough. The resulting segmentation masks from each technique are compared with each other.

3.1. Threshold

One simple and straightforward way to convert the experimental images into a binary segmentation mask is by using a pixel intensity threshold. Performed in [38] with the image processing toolbox, a threshold is chosen by using Otsu's method that uses discriminant analysis to pick the optimal threshold for the best separation of objects from their background [39]. The resulting images are converted to grayscale and each pixel falls below or above the threshold, converting

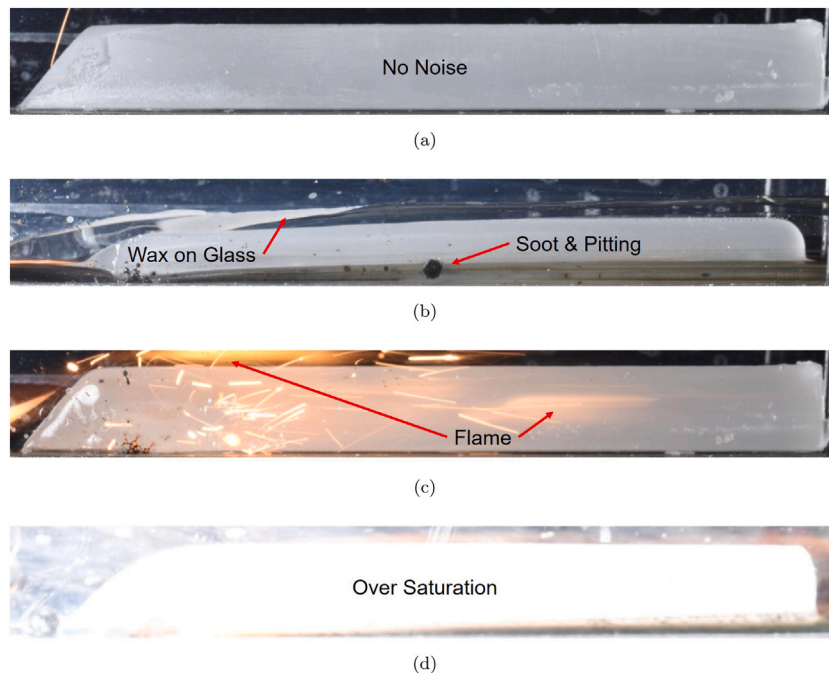


Fig. 3. Examples of side wax profile images using a high intensity flash from [5] with (a) no noise and noise from (b) wax on the glass, soot, pitting, (c) flame, and (d) over saturation pointed out.



Fig. 4. An example of the masks of the regressing fuel with the profile height tracking. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

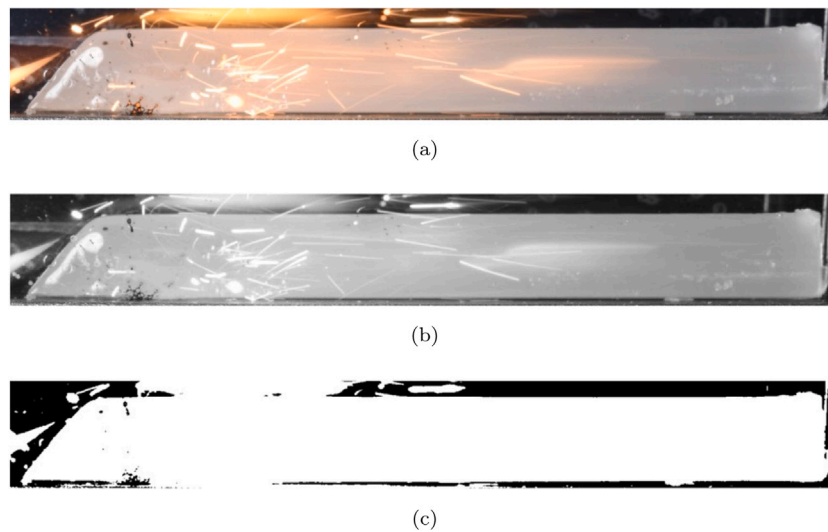


Fig. 5. An example of the process of the thresholding method starting from top to bottom, (a) original, (b) grayscale, and (c) segmented images.

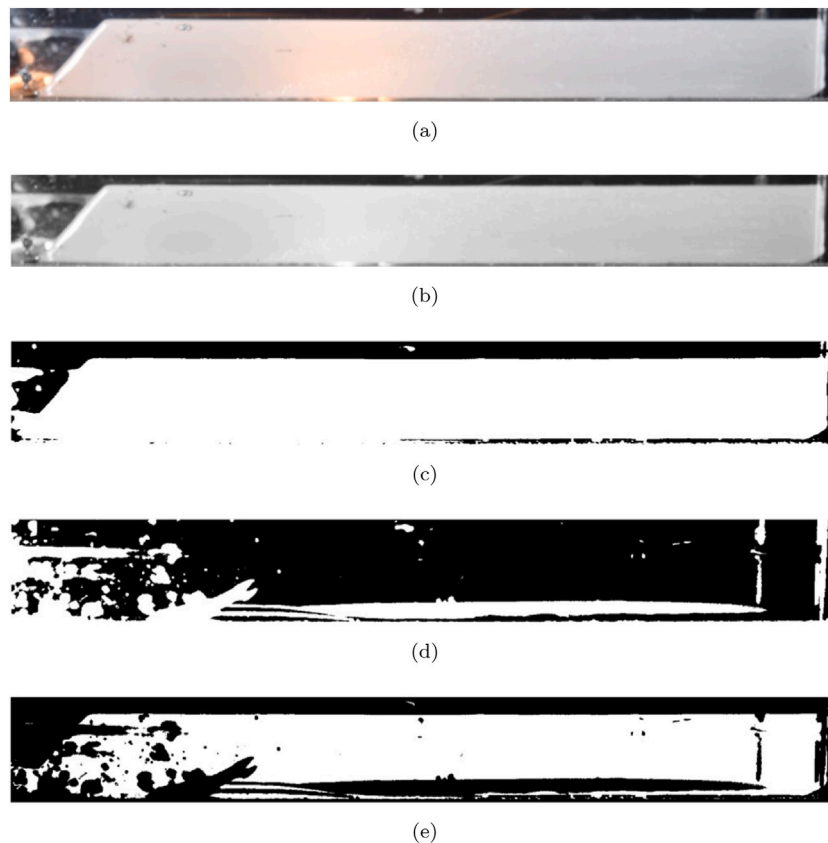


Fig. 6. An example of the process of the TLIS method starting from top to bottom, (a) original, (b) grayscale, (c) original segmented, (d) last segmented, and (e) original segmented minus the last segmented images.

its pixel value to a “0” or “1”, respectively. The grayscale images are converted by using the weighted sum formula $0.2989R + 0.5870G + 0.1140B$ [40], where R , G , and B are the corresponding red, green, and blue channels, respectively (MATLAB’s standard method to convert to grayscale). Pixels that have a value of “0” correspond to background, and pixels that have a value of “1” correspond to the fuel area. An example of this process is shown in Fig. 5. The thresholding method is very easy to quickly apply and provides good results if there is little to no noise. However, if there is significant noise (such as the data used

in this paper), this method will classify it as fuel and cause high error regression rate estimates.

3.2. Threshold Last Image Subtraction (TLIS)

The TLIS technique reduces the noise that affects the mask data. The last image in the series of images is assumed to have the maximum noise originating from pitting/soot/wax on the glass. Thus, in TLIS this last image is subtracted from the rest of the images’ binary masks. TLIS is helpful because it removes the locations with high amounts of noise,

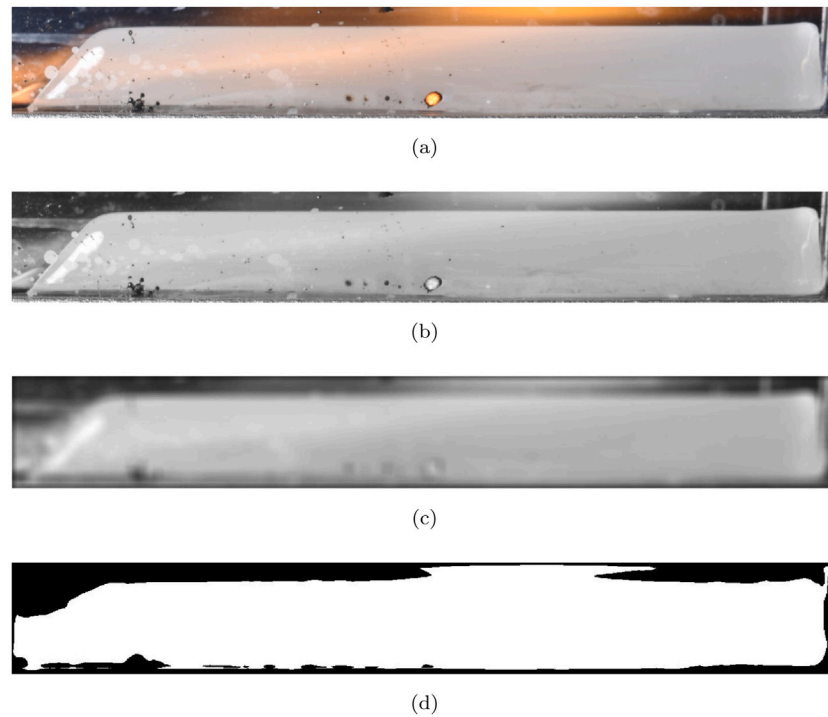


Fig. 7. An example of the process of the spatial filtering method starting from top to bottom, (a) original, (b) grayscale, (c) spatially filtered, and (d) segmented images.

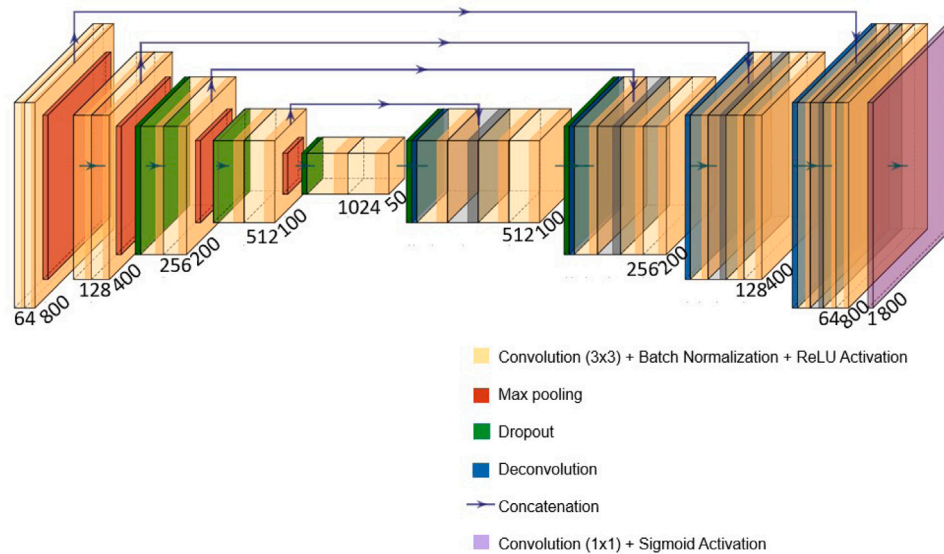


Fig. 8. U-net neural network architecture with dropout in the intermediate layers based on [21] and [37]. The input images are 800 by 800 by 3, their size is reduced and their depth increased in the contraction path of the U-net, and the opposite process takes place in the expansion path to arrive at the output binary segmentation mask of size 800 by 800 by 1.

to minimize the errors in the calculation of regression rate. An example of this process is shown in Fig. 6. TLIS is very simple and relatively easy to apply and does not classify any noise as fuel. However, this method reduces the amount of data significantly and causes error in the regression rate estimates.

3.3. Spatial filtering

Another image processing method that helps reduce the amount of noise that affects the data, specifically by filtering out the small data clusters, is spatial filtering. The images are first reduced in resolution by 50% (~300 by 2100) and then a 30 by 30 (~0.1% of the original

image) sized square block sliding average filter is applied. The data is reduced to enable, effectively, a larger window without being unreasonably computationally expensive. A larger window could have been chosen, but it is at the cost of more computational resources and at the increasing cost of the definition of the fuel boundaries. An example of this process is shown in Fig. 7.

3.4. U-net with Monte-Carlo dropout

The base U-net architecture was created using the Keras library [41]. For additional details on U-net architecture, readers are referred to [21]. U-net is the choice of network for this study because it is

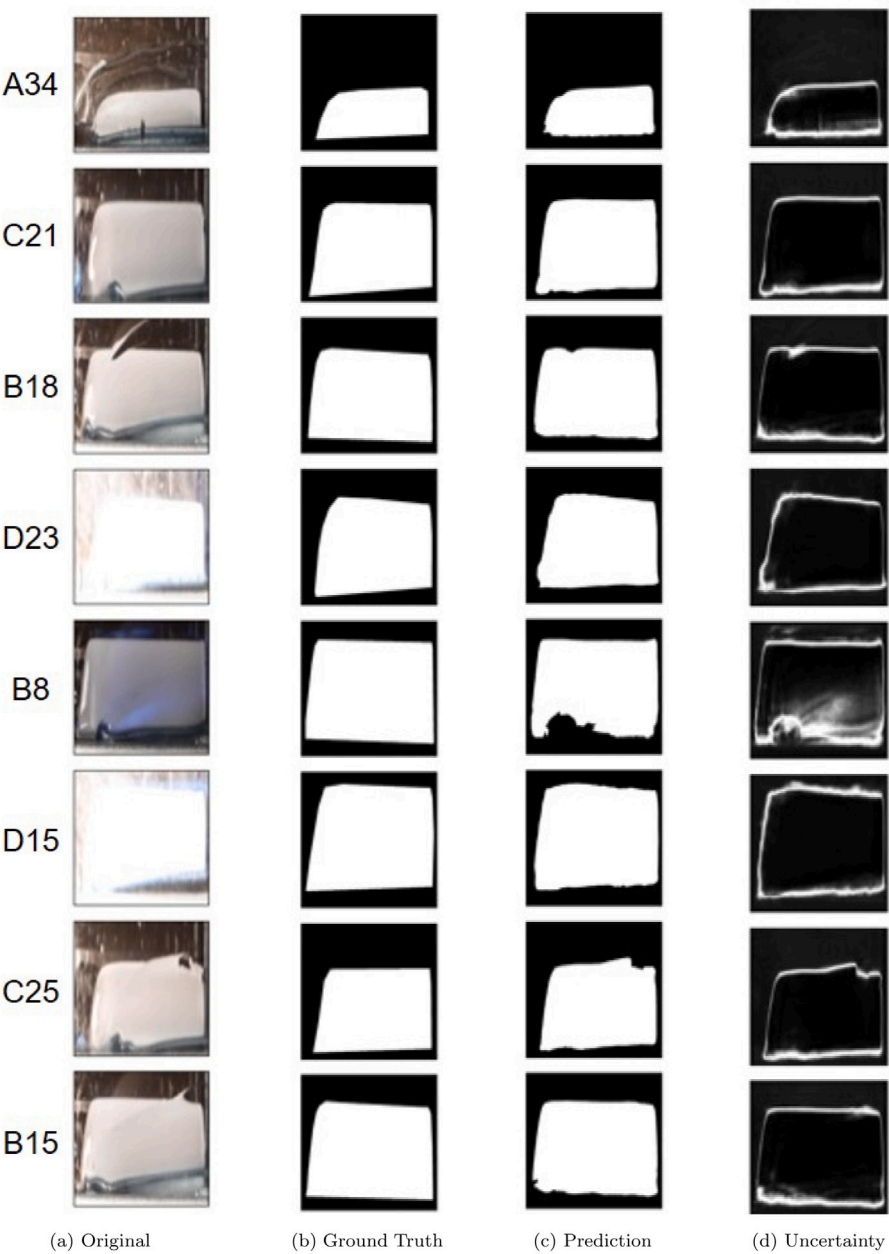


Fig. 9. Selected few of (a) the original images from the testing/validation set of various oxidizer fluxes, (b) the ground truth generated by the manual process, (c) the prediction from the trained neural network, and (d) the associated uncertainty map.

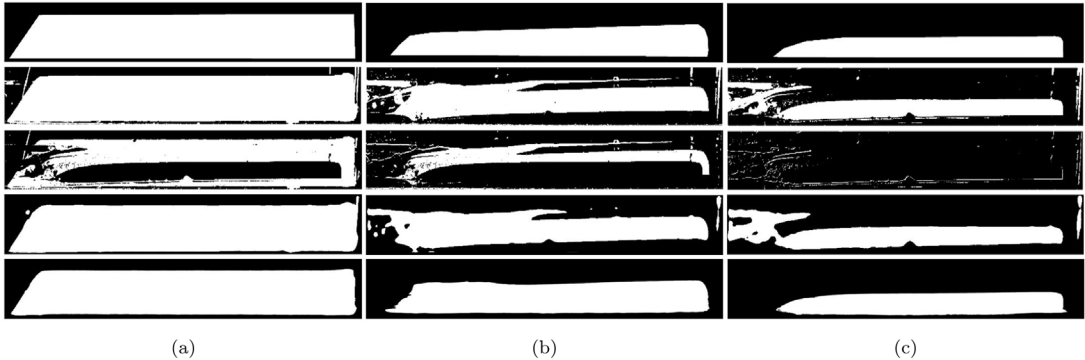


Fig. 10. From the top to bottom, ground truth, threshold, TLIS, spatial filtering, and the U-net binary masks for (a) early, (b) mid, and (c) late times for an oxidizer flux of 5.91 [kg/m²-s].

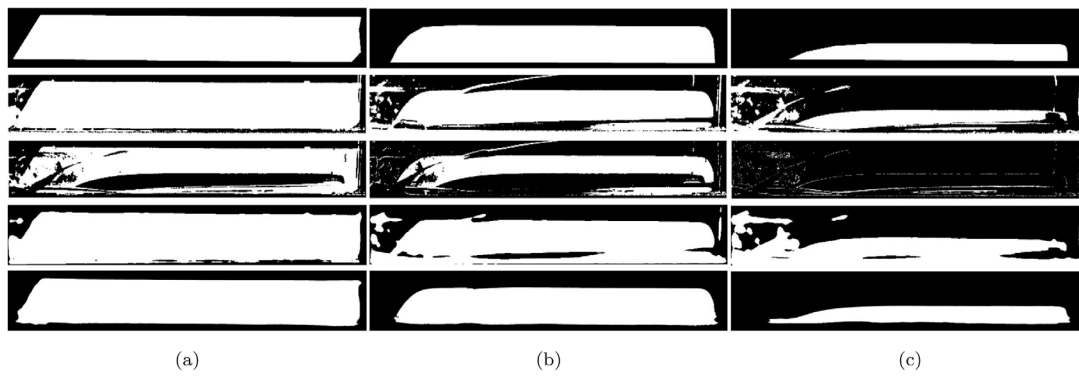


Fig. 11. From the top to bottom, ground truth, threshold, TLIS, spatial filtering, and the U-net binary masks for (a) early, (b) mid, and (c) late times for an oxidizer flux of 9.58 $[\text{kg}/\text{m}^2\cdot\text{s}]$.

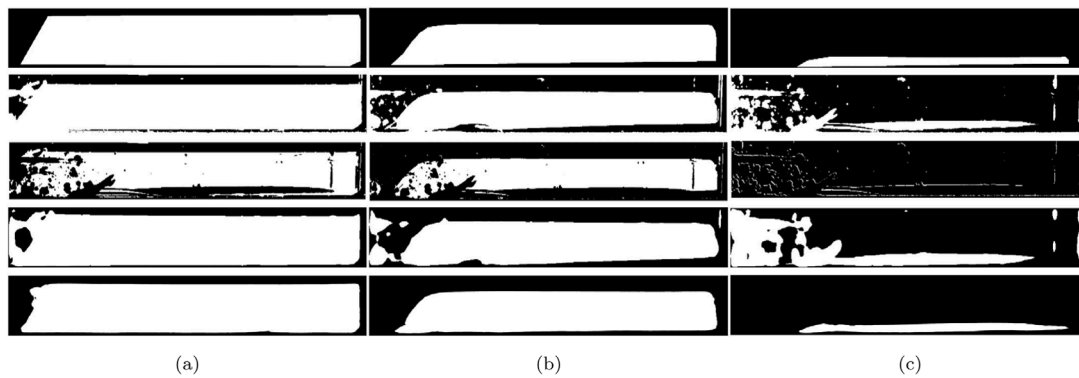


Fig. 12. From the top to bottom, ground truth, threshold, TLIS, spatial filtering, and the U-net binary masks for (a) early, (b) mid, and (c) late times for an oxidizer flux of 18.59 $[\text{kg}/\text{m}^2\cdot\text{s}]$.

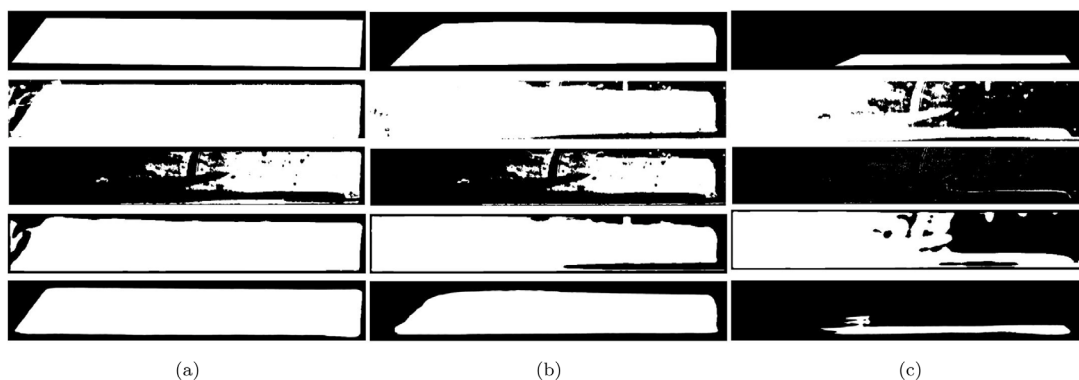


Fig. 13. From the top to bottom, ground truth, threshold, TLIS, spatial filtering, and the U-net binary masks for (a) early, (b) mid, and (c) late times for an oxidizer flux of 22.19 $[\text{kg}/\text{m}^2\cdot\text{s}]$.

capable of handling a low number of training data and still have a high success rate [21]. Fig. 8 shows a schematic of the U-net architecture used in this study. The deep network has a data reduction down-branch (contraction path) to locate the important features, followed by a data dilation up-branch (expansion path) to return to higher resolutions. U-net initially extracts the most important features from the image as resolution reduces and depth increases in the contraction path. Then, to acquire location information, the image is reconstructed by up-sampling to return to higher resolutions [21]. The contraction path consists of four sequences of two convolutions (3 by 3 kernels and ‘same’ padding) so that the output image has the same height/width as the input image), followed by batch normalization, ReLU activation, and a max pooling operation (2 by 2 kernel and strides equal to 2). The expansion path consists of four sequences of upsampling (2 by 2 kernels

and ‘same’ padding), followed by a concatenation with the feature map from the corresponding level in the contracting path, two convolutions (3 by 3 kernels and ‘same’ padding), batch normalization, and lastly a ReLU activation. Finally, U-net includes a last convolution operation (1 by 1 kernel, ‘same’ padding) to arrive at a gray-scale image of equal height and width as the original image (i.e., the fuel segmentation mask). The selected layer settings are consistent with those in [21] except for the extra padding in the convolutions to prevent the loss of border pixels, which is common practice [42]. Additionally, batch normalization is included which was not part of the original U-net paper) to reduce covariant shift, as suggested in [43].

To capture model uncertainty information (i.e. how confident the U-net is for each predicted pixel in the segmentation masks), the process outlined in [37] utilizing Monte-Carlo Dropout is followed to generate

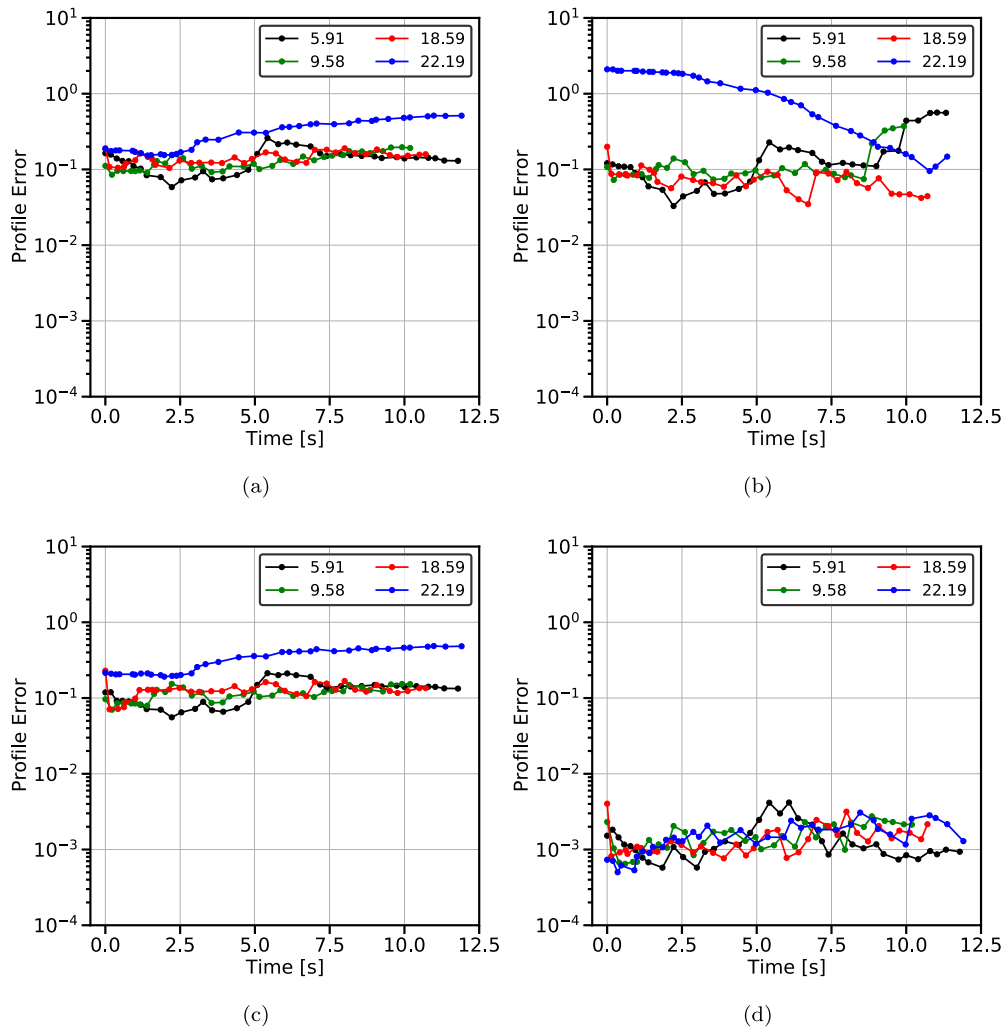


Fig. 14. Relative spatial error between the ground truth mask profiles and the profiles from (a) the threshold, (b) TLIS, (c) spatial filtering, and (d) U-net.

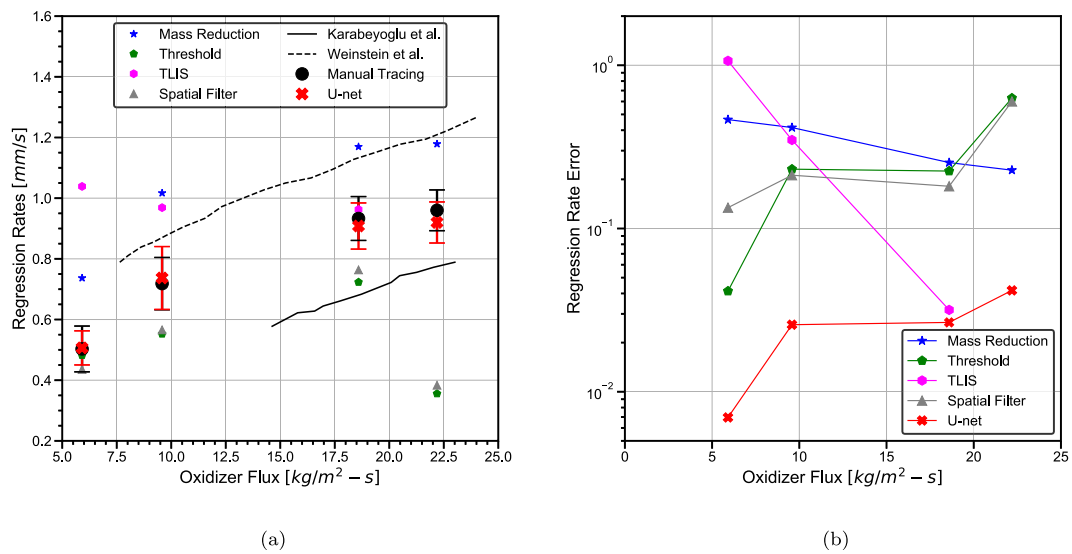


Fig. 15. (a) Comparison to published data from [5] and (b) the absolute error for the imaging processing techniques assuming the manual data from [5] as truth (TLIS did not give a feasible result for 22.19 [kg/m²-s] and therefore is not shown).

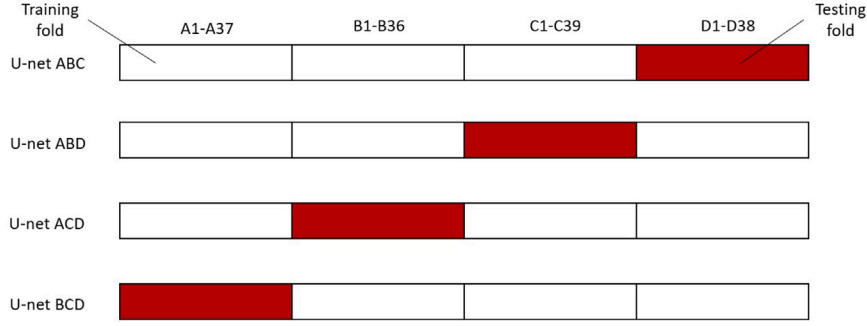


Fig. 16. Leave-one-out cross-validation (LOOCV) process. In total, four different U-net models were built by using images from 3 out of 4 oxidizer fluxes as the training set each time, and images from the left-out oxidizer flux as the testing set. Image naming convention follows Table 1.

the associated uncertainty map for each predicted segmentation mask. There are other approaches to quantify uncertainty in CNNs (e.g., using Bayesian networks that use distributions for the model weights [44]), but Monte Carlo Dropout is selected since the dropout layers can be easily implemented as extensions to the original U-net architecture. First, the U-net is trained with a 50% dropout probability in the intermediate layers to reduce overfitting according to [45]. The dropout process randomly silences neurons in the intermediate layers of the U-net by zero-ing the weights according to the set probability. Then at inference, the trained model is sampled 20 times [37], and the predicted segmentation masks are computed as the mean masks of the sampled model instances. The uncertainty map is then calculated as the entropy of the averaged probability vectors for the two class dimensions (“fuel” and “noise”), as described in [46, p. 309]:

$$U = - \sum_{c=1}^2 p_c \log(p_c) = -[p_{fuel} \log(p_{fuel}) + p_{noise} \log(p_{noise})] \quad (1)$$

where U is the uncertainty map, p_{fuel} is the predicted probability that the pixel corresponds to fuel, and p_{noise} is the predicted probability that the pixel corresponds to noise/background.

In order to increase robustness and reduce biases, the training data is initially augmented. Using the Albumentations library [47], each training image is rotated and flipped in random directions and magnitudes before each epoch. The total number of images after using the data augmentation is the same as before the augmentation, there are no additional artificial images. As a result, the training images are almost entirely augmented images as the chance of the network using original images (without flips and rotations) is very low but still possible. The most common loss function for binary image segmentation—binary cross entropy loss function, is used here in combination with a sigmoid activation layer. The binary cross entropy loss function is defined as,

$$\ell(x, y) = \overline{\{l_1, \dots, l_N\}^T}, \quad (2)$$

where ℓ is the loss function, the overbar corresponds to the mean all l_n , and

$$l_n = y_n \cdot \log \sigma(x_n) + (1 - y_n) \cdot \log(1 - \sigma(x_n)) \quad (3)$$

where, y_n is the prediction, σ is the sigmoid activation function, and x_n is the ground truth. Finally, the Adam optimization algorithm is used to update the weights in the kernels. The algorithm is chosen for its ability to handle noisy data [48,49] instead of stochastic gradient descent that was used in the original U-net paper.

4. Results and discussion

4.1. Training on all available images

As an initial test, the 150 RGB images are split into the training set (80% or 120 images) and the testing and validation set (20% or 30 images). The images are randomly assigned a pseudorandom number

Table 1

Image naming conventions with the letters corresponding to different oxidizer fluxes and the numbers corresponding to the chronological order that the images were captured at.

Oxidizer flux, G [kg/m ² -s]	First captured image	Last captured image	Total images
5.91	A1	A37	37
9.58	B1	B36	36
18.59	C1	C39	39
22.19	D1	D38	38

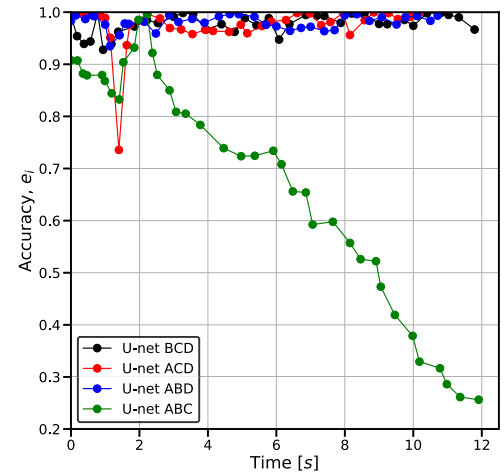


Fig. 17. Leave-one-out cross-validation (LOOCV) accuracy measure for all images in the testing oxidizer flux, corresponding to different experiment burn times.

Table 2

Confusion matrix for output of U-net predictions. n represents the corresponding number of pixels.

	Predicted: Fuel	Predicted: Noise
Truth: Fuel	n_1	n_2
Truth: Noise	n_3	n_4

generator to the training or validation sets, so that information from all four fluxes are used both in training the U-net to correctly capture the experiment phenomena and validate the U-net model to test whether it can reliably be used for fuel regression calculation. To train and validate the U-net, the original experiment images and manually traced masks are resized to 800 by 800 pixels to limit computational expense. An aspect ratio of 1 ensures valid matrix operations during the convolution and deconvolution operations in the deeper layers on the U-net. The 150 RGB experimental images follow a naming convention based on the oxidizer flux and the time of the experiment they are captured shown in Table 1.

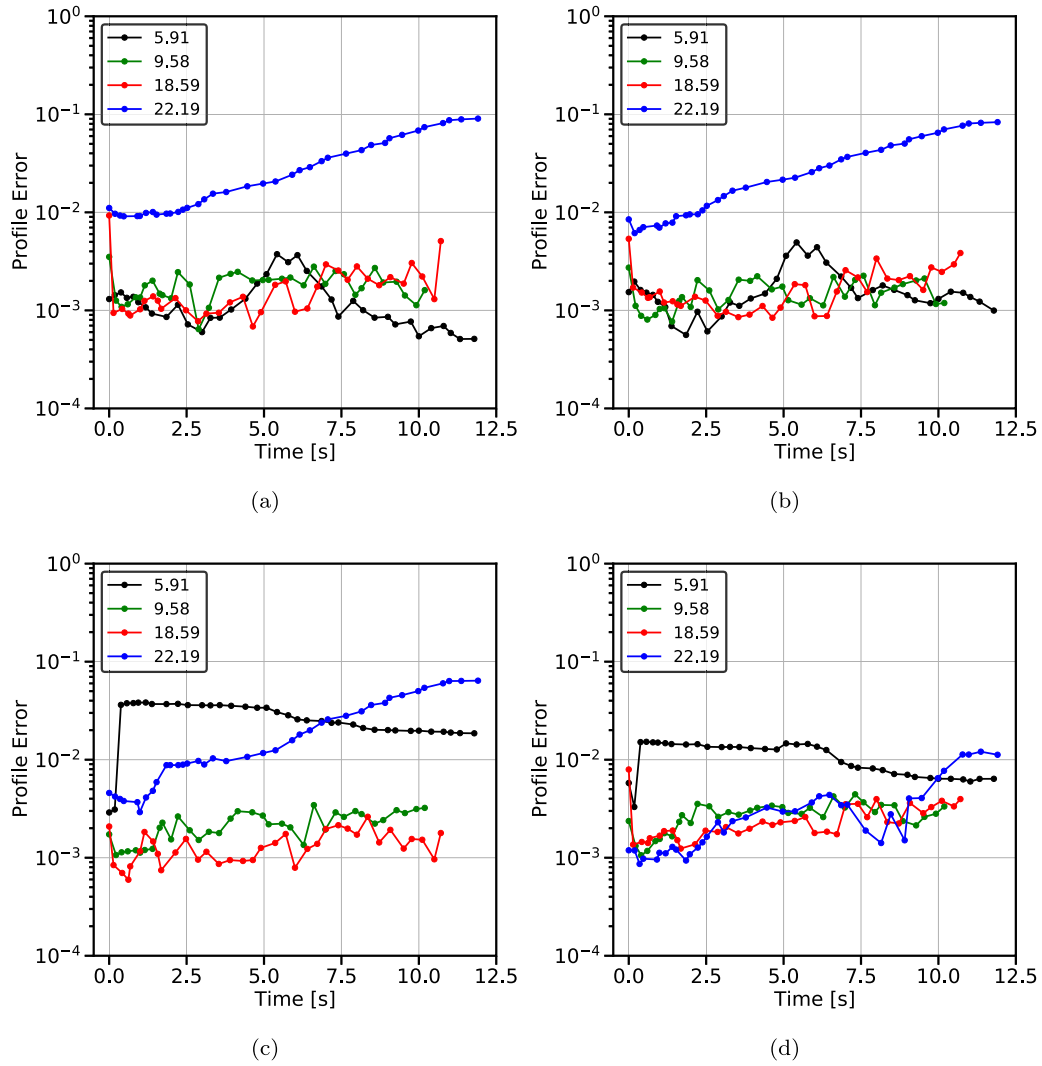


Fig. 18. Spatial error between the ground truth mask profiles and the predicted profiles from U-net (a) trained on only 5.91, (b) trained on only 9.58, (c) trained on only 18.59, and (d) trained on only 22.19 $[\text{kg}/\text{m}^2\text{-s}]$ RGB images.

For the neural network parameters, the batch size was set to 5 to account for possible model degradation in its ability to generalize due to the small training set as shown by [50] and the starting learning rate is 10^{-3} , the default setting for the Adam optimizer (the learning rate is adapted during the training process). For this initial training, the U-net required 6 GB of memory, 3 cores, and runtime of 20.5 h.

A selected few examples of original resized images of the testing set are shown in Fig. 9a, the associated ground truth binary masks manually traced after the resolution reduction are shown in Fig. 9b, the neural network's predictions are shown in Fig. 9c, and the associated uncertainty maps are shown in Fig. 9d.

The U-net model trained from images of all fluxes visually predicted the segmentation masks successfully. For example, for images A34, B18, and B15, U-net correctly traces the fuel mask without including the melted wax on the glass of the image. It also correctly predicts the mask for the over-saturated images D15 and D23. Some abnormal geometries (e.g., B8 and C25) are found to be the hardest to predict correctly. The uncertainty maps showed the largest level of uncertainty to be on the fuel boundaries and around abnormal geometries in the predicted masks (e.g., B8). The model uncertainties inside and outside the identified fuel mask border are very small, showing that once U-net identified the border of the fuel masks with some uncertainty, it correctly learned to classify the pixels inside and outside the border. Lastly, U-net performed well for fuel mask predictions in all stages

of the experiment. Reliable prediction of the fuel masks over time is an important feature if U-net is to be used for fuel regression rate measurements because measurement requires tracking of the fuel mask over the entire burn duration.

Predictions from the oxidizer fluxes of 5.91, 9.58, 18.59, and 22.19 $[\text{kg}/\text{m}^2\text{-s}]$ using threshold, TLIS, spatial filtering, and U-net at early ($t \approx 0$ s), mid ($t \approx 5$ s), and late ($t \approx 10$ s) times are presented in Figs. 10–13. Visually, the U-net predictions to perform the best and the simple image processing methods especially struggle with the oxidizer flux of 22.19 $[\text{kg}/\text{m}^2\text{-s}]$ case in Fig. 13 most likely due to the over-saturated data seen uniquely in only the 22.19 $[\text{kg}/\text{m}^2\text{-s}]$ case. The threshold, TLIS, and spatial filtering all have some advantages and do provide reasonable predictions, however, they do not provide the level of reliable filtering that U-net provides.

Fig. 14 shows the total relative spatial error, $(\sum |h - h_{\text{truth}}|/h_{\text{truth}}) / \text{width}_{\text{image}}$ between the ground truth mask profiles and the profiles from the image processing techniques for each point in time ($\text{width}_{\text{image}}$ is the sum of the number of pixels in the image in the horizontal direction). The U-net results are resized into the native resolution of the ground truth images in order to compare the profiles. Overall, the U-net produced the lowest maximum error out of all four image processing methods. The U-net architecture is the only approach that performed the most consistent out of the other image processing methods with error below 10^{-2} for all fluxes. Specifically, in the 22.19 $[\text{kg}/\text{m}^2\text{-s}]$

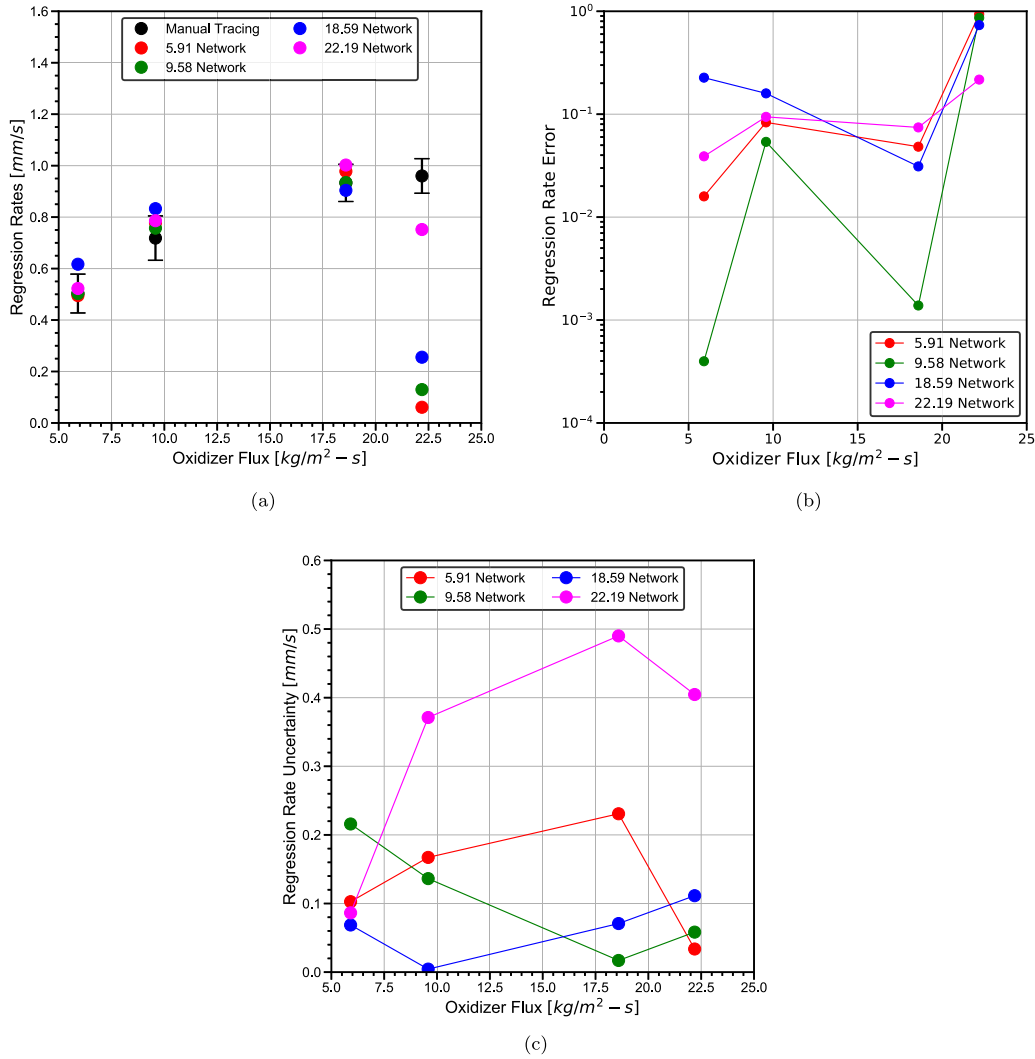


Fig. 19. (a) Regression rate measurements based on fuel mask predictions from the individual oxidizer flux U-net networks and comparison to published data from [5], (b) the absolute error assuming the manual data from [5] as truth, and (c) the corresponding uncertainty in the fuel profile.

case, the U-net filtered the noise caused by over-saturation correctly, whereas, the other image processing methods generated significant errors, especially the TLIS method with errors over 10^0 .

Fig. 15a shows the comparison between the regression rate obtained from the different image processing techniques and the ground truth (manually drawn data from [5]). The results from the manual process are in black circles and the results from U-net are shown by the red circles. The regression rate error bars for the U-net approach are calculated by first getting an average threshold value for each mask by summing the entire uncertainty in the map and then normalizing by the number of pixels in the mask. If a pixel's uncertainty falls below this threshold the pixel becomes a "0 (noise)" and if it falls above, it becomes a "1 (fuel)". This process creates another binary mask which is added and subtracted to the original mean binary masks to get two new sets of regression rate data. The maximum deviation from the original regression rate from the two new regression rates is then calculated and this value is used for the error bars. The regression rate measurements inferred from the measurement fuel mass before and after the burn are labeled as "Mass Reduction". The results from the other image processing methods are also plotted. The TLIS method did not provide feasible results for the data for the oxidizer flux of 22.19 [kg/m²-s] resulting in no data for this method at this oxidizer flux. The black dashed and solid lines are regression rates obtained by [51] and [52], respectively. Fig. 15b shows the total absolute error,

$|\dot{r}_{prediction} - \dot{r}_{truth}|/\dot{r}_{truth}$, assuming the manual tracing, the black circles in Fig. 15a, from [5] as truth. TLIS performs poorly for 5.91 and 9.58 [kg/m²-s] fluxes, but performs well for 18.59 [kg/m²-s] with an error below 10%. As for the 22.19 [kg/m²-s] flux, the TLIS results do not provide a feasible regression rate and therefore is not shown. The threshold and spatial filtering have approximately the same accuracy as the mass reduction method. Almost all of the regression rate data points from all of the baseline image processing methods do not fall in the error bars for the manual tracing. The U-net regression rate estimates agree well and are all inside the manual tracing error bars with regression rate error for all fluxes below 10%. The U-net error bars as well encompass the manual tracing data points. U-net is the only method that consistently predicts the regression rate for all oxidizer fluxes.

4.2. U-net cross-validation

The results of the U-net architecture presented in the previous section come from training the model with data from all available oxidizer fluxes and showed that U-net accurately predicts the fuel regression rate. However, the image quality varies significantly across the four oxidizer fluxes (e.g., images from 22.19 [kg/m²-s] are over-saturated as shown in Fig. 9). Therefore, to further validate the U-net architecture and its capability to make accurate predictions on data

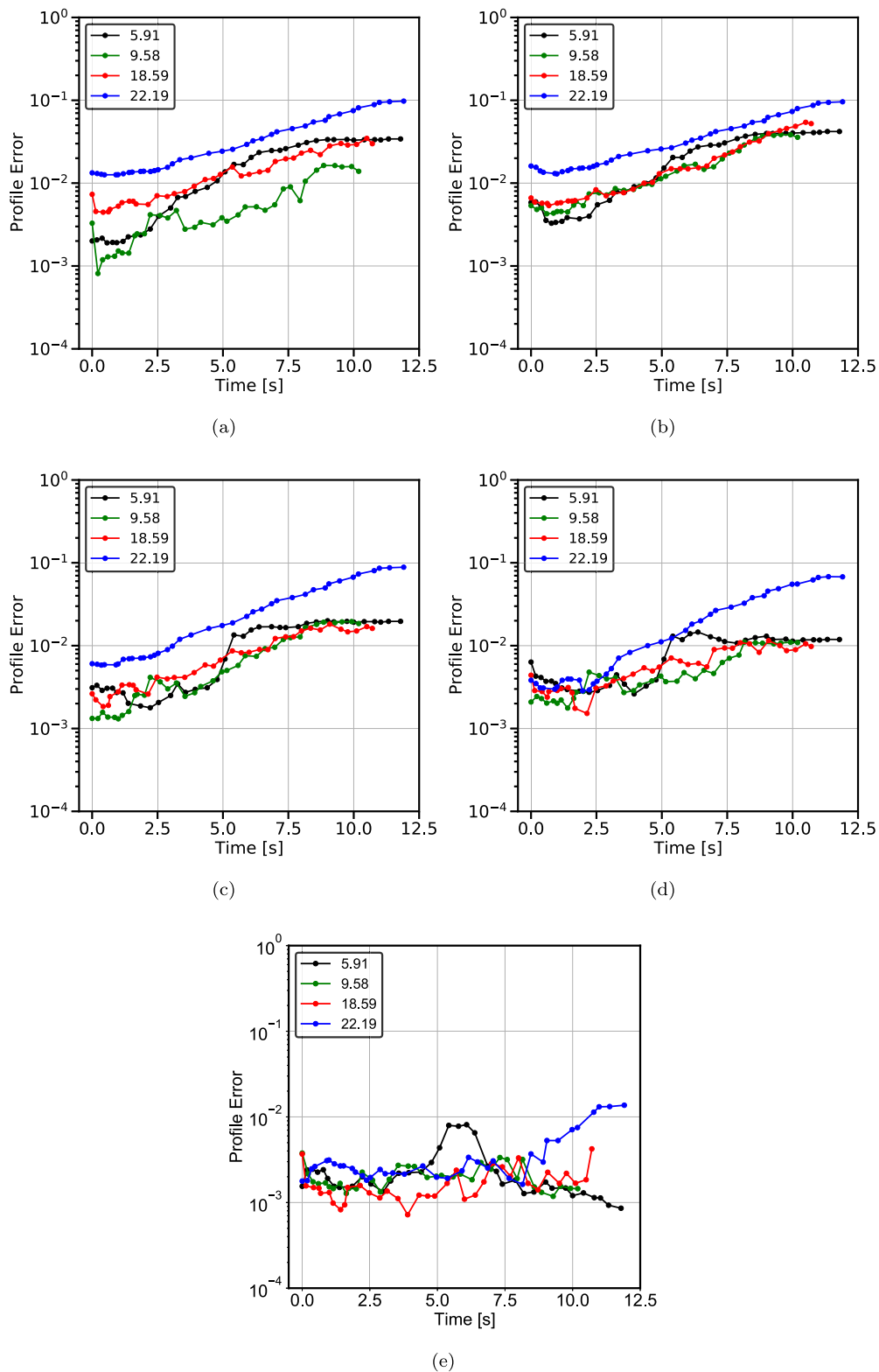


Fig. 20. Relative spatial error between the ground truth mask profiles and the predicted profiles from U-net (a) trained on only 5.91 [kg/m²-s], (b) trained on only 9.58 [kg/m²-s], (c) trained on only 18.59 [kg/m²-s], (d) trained on only 22.19 [kg/m²-s], and (e) trained on all fluxes, using grayscale images.

from different fluxes, a Leave-one-out cross-validation (LOOCV) [53] approach is implemented. To complete the cross-validation process, the dataset is initially split into 4 folds, one for each oxidizer flux. Then, a U-net model is built by using 3 of the folds as the training set and the last fold as the testing set [54] (i.e., train with images from 3 out of

4 oxidizer fluxes and test on images from the left-out flux as shown in Fig. 16).

Each of the U-net models created during the cross-validation are evaluated based on an accuracy metric that represents the number of correctly classified pixels as fuel or noise for all images in the

corresponding testing fold. For each testing image, a confusion matrix is created (see Table 2).

The accuracy measure is the ratio of correctly classified pixels by the model in the particular testing set, over the total number of pixels:

$$e_i = \frac{n_{correct}}{n_{total}} = \frac{n_1 + n_4}{\sum_{i=1}^4 n_i} \quad (4)$$

Following the validation process, the results are shown in Fig. 17.

The U-net architecture predicts the experimental images for the oxidizer fluxes 5.91, 9.58, and 18.59 [kg/m²-s] correctly with an accuracy >95%. However, the model that is trained without the over-saturated images of the 22.19 [kg/m²-s] oxidizer flux (U-net ABC) fails to make correct predictions past the initial phases of the experiment. The result indicates that when U-net has not learned features of images with high noise or over-saturation, it should not be used for predicting fuel segmentation masks. Including images with high noise, improves the predictive power of the architecture and in those cases, U-net is a viable option for predicting fuel segmentation masks for fuel regression rate measurement.

The U-net ACD does not predict one particular image well, B8. The image includes large amounts of noise from the bottom plate of the chamber and the model fails to classify that area as fuel. Even when trained from data on all four fluxes, noise in B8 is not captured correctly by the U-net, as shown previously in Fig. 9.

After training and validating the U-net architecture using all the available images from the experiments, two practical applications of fuel regression rate measurement are investigated in the following sections: (A) the availability of experimental images from only one oxidizer flux and (B) the availability of only grayscale images. These two scenarios may be frequent in fuel regression measurement research either due to laboratory equipment constraints, process data that has not been manually processed, or an inability to repeat or collect measurements.

4.3. Application A: Oxidizer flux independency

The ability of the neural network to make satisfactory fuel mask predictions independent of the oxidizer flux is investigated by training the network on only one flux data set and testing on the remaining three. This application is particularly useful to explore when the data for other oxidizer fluxes is to be processed without having access to the manually traced fuel masks. For these cases, due to the low number of training images and to avoid overfitting, an additional stop criterion is implemented: model training should stop when the model does not improve at predicting the remaining three fluxes for 20 consecutive epochs. In total, four separately trained neural networks are produced and are used to predict all of the other oxidizer fluxes. Fig. 18 shows the total relative spatial error, $(\sum |h - h_{truth}|/h_{truth})/width_{image}$ between the ground truth mask profiles and the profiles from each network's prediction for each point in time ($width_{image}$ is the sum of the number of pixels in the image in the horizontal direction). Overall, each network outperforms each of the basic image processing techniques. The networks trained with the images from fluxes 5.91, 9.58, and 18.59 [kg/m²-s], predict the 22.19 [kg/m²-s] case with the largest error ($>10^{-2}$), particularly during the later stages of the fuel burn, most likely due to the presence of over-saturated images. However, the model trained on the data from the 22.19 [kg/m²-s] flux (Fig. 18d) is overall better at capturing the profile for all the fluxes, as the profile error for each does not exceed 10^{-2} in any case.

Fig. 19a shows the measurement of regression rates from the predicted fuel masks for each of the four networks. Fig. 19b shows the total absolute error, $|\dot{r}_{prediction} - \dot{r}_{truth}|/\dot{r}_{truth}$, assuming the manual tracing (represented as the black circles) from [5] as truth. Fig. 19c shows the uncertainty for each network, derived from the uncertainty map of each predicted fuel mask profile. The networks for 5.91, 9.58, and 18.59

[kg/m²-s] predict the regression rate within first order accuracy for each other, but do not have satisfactory performance at predicting the regression rate for the 22.19 [kg/m²-s] case. The network trained with the 22.19 [kg/m²-s] images, has approximately a first order accuracy for all of the oxidizer fluxes as discussed before, however, it also has the overall highest uncertainty in the model. With monochrome data, it is possible to get good regression rate predictions, however order to get sufficiently accurate results the networks needs more training data. In such scenarios, the use of a high speed camera should provide enough data to train the network and obtain improved regression rate estimates.

4.4. Application B: Monochrome study

To further explore the limits of satisfactory fuel mask predictions with the U-net, a study is performed on how the network performs when trained on lower depth image data. This study is useful in applications where only a monochrome camera is available, which costs a fraction compared to a color high speed camera. In this case, the experimental RGB images are first converted to grayscale images before using them as the training set. Similar to the individual flux cases, to avoid overfitting due to the small training set, the additional stop criterion to halt model training if it has not improved at prediction for 20 consecutive epochs is implemented. Fig. 20 shows the total relative spatial error, $(\sum |h - h_{truth}|/h_{truth})/width_{image}$ between the ground truth masks profiles and the profiles from each network's prediction for each point in time ($width_{image}$ is the sum of the number of pixels in the image in the horizontal direction). The networks trained with the 5.19, 9.58, and 18.59 [kg/m²-s] grayscale images, have the highest error when predicting fuel masks for the 22.19 [kg/m²-s] case particularly at the later stages of fuel burn (shown in Figs. 20a to 20c). The network trained with the 22.19 [kg/m²-s] grayscale images, also has the highest error when predicting fuel masks for the 22.19 [kg/m²-s] case and after inspection of the results (shown in Fig. 20d and an example of the predicted mask is shown in Fig. 21), there is significant noise most likely caused by the over-saturation in the experimental images for the 22.19 [kg/m²-s] flux that are problematic for the network predictions. At the later stages of the burn, more noise is present as the combustion process adds more deposits of wax, soot, and pitting on the glass of the chamber, creating the increasing error as time advances trend seen in Figs. 20a to 20d. The network that is trained on all fluxes uses images that are randomly chosen using a pseudorandom number generator (similar to Section 4.1) out of the 150 total images. The all flux network has the lowest error $< 10^{-2}$ for all fluxes and experimental times (shown in Fig. 20e). For the all flux case, the training size is roughly 4 times larger (120 images) compared to the individual flux cases (about 30 images), that is expected to reduce the error in combination with a better generalization of the training data. Additionally, the all flux network performs more consistently at late experimental times as compared to the other networks, suggesting that the training size of the individual flux network may be too small.

Fig. 22a shows the comparison between the regression rate obtained from the networks trained on grayscale images from different oxidizer fluxes and the ground truth (manually drawn data from [5]). Fig. 22b shows the total absolute error, $|\dot{r}_{prediction} - \dot{r}_{truth}|/\dot{r}_{truth}$, assuming the manual tracing (represented as the black circles), from [5] as truth. Fig. 22c shows the uncertainty for each network. The results for the networks trained on grayscale images from 5.91, 9.58, and 18.59 [kg/m²-s] all have over first order error and do not provide good regression rate results consistently. The network trained on 22.19 [kg/m²-s] provides the best results out of the four oxidizer flux independently trained networks, however falls short on predicting the regression rate for its own data, likely due to the noise seen in Fig. 21c. Even the network trained on all the oxidizer fluxes has better results predicting fuel masks for 5.91, 9.58, and 18.59 [kg/m²-s] with error less than 10^{-1} but still has difficulty predicting the 22.19 [kg/m²-s] case. The networks trained with lower depth data, as expected, have higher error in predicting the regression rate compared to the ones trained with RGB data.

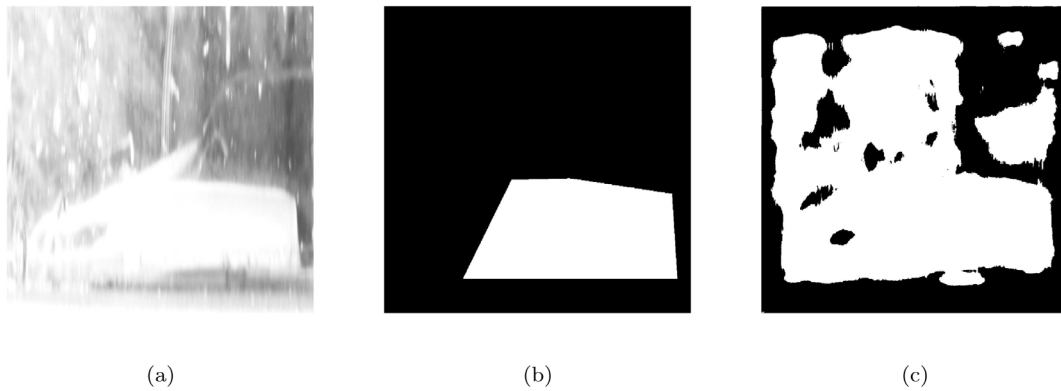


Fig. 21. An example of data from the 22.19 [kg/m²·s] network predicting the 22.19 [kg/m²·s] case where (a) is the resized grayscale image, (b) is the ground truth, and (c) is the fuel mask prediction.

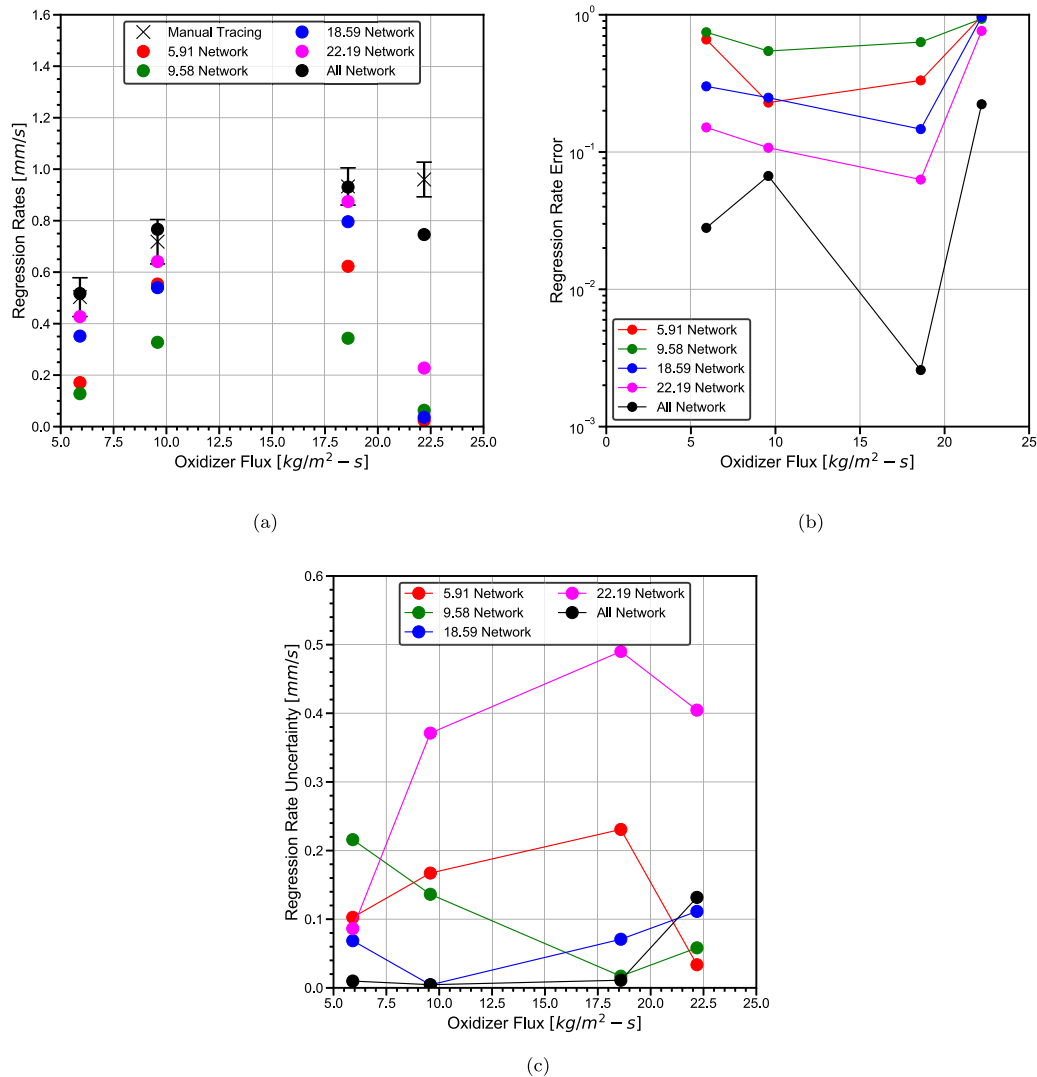


Fig. 22. (a) Grayscale network regression rate results comparison to published data from [5], (b) the absolute error assuming the manual data from [5] as truth, and (c) the corresponding uncertainty.

5. Conclusion

In this study, for the first time, a CNN architecture, specifically, U-net is applied to measure the solid fuel regression rate in a 2D slab burner hybrid rocket combustion experiment. The slab burner

experiment consists of gaseous oxygen flowing through a rectangular combustion chamber with optical access on both sides. Paraffin wax is ignited in the chamber and is photographed continuously with a high intensity flash. The resulting images are high in noise caused by interference from wax, soot, pitting, and flame. Four different image

segmentation techniques are applied and resulting regression rate error estimates are quantified. The U-net neural network is found to perform the best and the regression rate estimates are in very good agreement with the values from literature. The uncertainty in regression rate originating from the U-net architecture is quantified using a Monte Carlo Dropout method. The U-net architecture training is independent of the oxidizer mass flux and provides good estimates, even for noisy data. The U-net network performs well in situations where the data can be sparse, such as with monochrome images but the data must have a minimum level of detail to which features can be detected and patterns can be found. In situations where the data is overwhelmingly abundant, such as having access to high speed imagery, the authors recommend choosing a training set that is a uniformly distributed subset based on experiment times (e.g., before ignition to capture the full fuel profile, early stages after ignition to capture early combustion, middle stages to capture steady state flame, and late stages to capture images with high noise). Such selection would improve the network's ability to generalize well to images and fluxes outside the training data. Overall, the U-net architecture and proposed techniques appear to successfully automate manual fuel mask segmentation and provide accurate estimates of the regression rate in slab burner experiments for hybrid rockets.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Funding: This work was supported by the United States Department of Energy's (DoE) National Nuclear Security Administration (NNSA) under the Predictive Science Academic Alliance Program III (PSAAP III) at the University at Buffalo, under contract number DE-NA0003961.

Computational resources: The authors acknowledge the Tufts University High Performance Compute Cluster (<https://it.tufts.edu/high-performance-computing>) which was used for training and evaluating the U-net models.

References

- [1] E.T. Jens, B.J. Cantwell, G.S. Hubbard, Hybrid rocket propulsion systems for outer planet exploration missions, *Acta Astronaut.* 128 (2016) 119–130, <http://dx.doi.org/10.1016/j.actaastro.2016.06.036>.
- [2] G. Gallo, S. Mungiguerra, R. Savino, New entrainment model for modelling the regression rate in hybrid rocket engines, *J. Propuls. Power* (2021) <http://dx.doi.org/10.2514/6.2021-138333>.
- [3] M.J. Chiaverini, N. Serin, D.K. Johnson, Y.-C. Lu, K.K. Kuo, G.A. Risha, Regression rate behavior of hybrid rocket solid fuels, *J. Propuls. Power* 16 (1) (2000) 125–132, <http://dx.doi.org/10.2514/2.5541>.
- [4] M.A. Karabeyoglu, *Transient Combustion in Hybrid Rockets* (Ph.D. thesis), Stanford University, 1998.
- [5] K. Budzinski, S.S. Aphale, E.K. Ismael, G. Surina, P.E. DesJardin, Radiation heat transfer in ablating boundary layer combustion theory used for hybrid rocket motor analysis, *Combust. Flame* 217 (2020) 248–261, <http://dx.doi.org/10.1016/j.combustflame.2020.04.011>.
- [6] G. Ziliac, M. Karabeyoglu, Hybrid rocket fuel regression rate data and modeling, in: 42nd AIAA/ASME/SAE/ASEE Jt. Propuls. Conf. Exhib., AIAA, 2006, <http://dx.doi.org/10.2514/6.2006-4504>.
- [7] Y. Hirata, S. Aso, T. Hayashida, R. Nakawata, Y. Tani, K. Morishita, T. Shimada, Improvement of regression rate and combustion efficiency of high density polyethylene fuel and paraffin fuel of hybrid rockets with multi-section swirl injection method, in: 47th AIAA/ASME/SAE/ASEE Jt. Propuls. Conf. Exhib., AIAA, 2011, <http://dx.doi.org/10.2514/6.2011-5907>.
- [8] C. Dunn, G. Gustafson, J. Edwards, T. Dunbrack, C. Johansen, Spatially and temporally resolved regression rate measurements for the combustion of paraffin wax for hybrid rocket motor applications, *Aerosp. Sci. Technol.* 72 (2018) 371–379, <http://dx.doi.org/10.1016/j.ast.2017.11.024>.
- [9] G. Cai, P. Zeng, X. Li, H. Tian, N. Yu, Scale effect of fuel regression rate in hybrid rocket motor, *Aerosp. Sci. Technol.* 24 (1) (2013) 141–146, <http://dx.doi.org/10.1016/j.ast.2011.11.001>.
- [10] W.H. Knuth, M.J. Chiaverini, J.A. Sauer, D.J. Gramer, Solid-fuel regression rate behavior of vortex hybrid rocket engines, *J. Propuls. Power* 18 (3) (2002) 600–609, <http://dx.doi.org/10.2514/2.5974>.
- [11] R. Kumar, P. Ramakrishna, Measurement of regression rate in hybrid rocket using combustion chamber pressure, *Acta Astronaut.* 103 (2014) 226–234, <http://dx.doi.org/10.1016/j.actaastro.2014.06.044>.
- [12] A. Karabeyoglu, G. Ziliac, B. Cantwell, S.D. Zilwa, P. Castellucci, Scale-up tests of high regression rate liquefying hybrid rocket fuels, in: 41st Aerosp. Sci. Meet. Exhib., AIAA, 2003, <http://dx.doi.org/10.2514/6.2003-1162>.
- [13] K.-H. Shin, C. Lee, S.Y. Chang, J.Y. Koo, The enhancement of regression rate of hybrid rocket fuel by various methods, in: 43rd AIAA Aerospace. Sci. Meet. Exhib., AIAA, 2005, <http://dx.doi.org/10.2514/6.2005-359>.
- [14] P. Korting, H. Schöyer, Y. Timmat, Advanced hybrid rocket motor experiments, *Acta Astronaut.* 15 (2) (1987) 97–104, [http://dx.doi.org/10.1016/0094-5765\(87\)90009-9](http://dx.doi.org/10.1016/0094-5765(87)90009-9).
- [15] C. Carmicino, A.R. Sorge, Influence of a conical axial injector on hybrid rocket performance, *J. Propuls. Power* 22 (5) (2006) 984–995, <http://dx.doi.org/10.2514/1.19528>.
- [16] M.J. Chiaverini, K.K. Kuo, A. Peretz, G.C. Harting, Regression-rate and heat-transfer correlations for hybrid rocket combustion, *J. Propuls. Power* 17 (1) (2001) 99–110, <http://dx.doi.org/10.2514/2.5714>.
- [17] S. Ghosh, A. Pal, S. Jaiswal, K.C. Santosh, N. Das, M. Nasipuri, SegFast-V2: Semantic image segmentation with less parameters in deep learning for autonomous driving, *Int. J. Mach. Learn. Cybern.* 10 (2019) 3145–3154, <http://dx.doi.org/10.1007/s13042-019-01005-5>.
- [18] C.L. Chowdhary, D. Acharya, Segmentation and feature extraction in medical imaging: A systematic review, *Procedia Comput. Sci.* 167 (2020) 26–36, <http://dx.doi.org/10.1016/j.procs.2020.03.179>.
- [19] S. Minaee, Y.Y. Boykov, F. Porikli, A.J. Plaza, N. Kehtarnavaz, D. Terzopoulos, Image segmentation using deep learning: A survey, *IEEE Trans. Pattern Anal. Mach. Intell.* (2021) <http://dx.doi.org/10.1109/TPAMI.2021.3059968>.
- [20] M. Yaqub, J. Feng, M.S. Zia, K. Arshid, K. Jia, Z.U. Rehman, A. Mehmood, State-of-the-art CNN optimizer for brain tumor segmentation in magnetic resonance images, *Brain Sci.* 10 (7) (2020) <http://dx.doi.org/10.3390/brainsci10070427>.
- [21] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, in: *Lecture Notes in Comput. Sci.*, Springer International Publishing, 2015, pp. 234–241, http://dx.doi.org/10.1007/978-3-319-24574-4_28.
- [22] S. Qamar, H. Jin, R. Zheng, P. Ahmad, M. Usama, A variant form of 3D-UNet for infant brain segmentation, *Future Gener. Comput. Syst.* 108 (2020) 613–623, <http://dx.doi.org/10.1016/j.future.2019.11.021>.
- [23] Z. Zeng, W. Xie, Y. Zhang, Y. Lu, RIC-UNet: An improved neural network based on unet for nuclei segmentation in histology images, *IEEE Access* 7 (2019) 21420–21428, <http://dx.doi.org/10.1109/access.2019.2896920>.
- [24] Y. Zhou, W. Huang, P. Dong, Y. Xia, S. Wang, D-UNet: A dimension-fusion u shape network for chronic stroke lesion segmentation, *IEEE/ACM Trans. Comput. Biol. Bioinform.* 18 (3) (2021) 940–950, <http://dx.doi.org/10.1109/tcb.2019.2939522>.
- [25] X. Li, H. Chen, X. Qi, Q. Dou, C.-W. Fu, P.-A. Heng, H-DenseUNet: Hybrid densely connected UNet for liver and tumor segmentation from CT volumes, *IEEE Trans. Med. Imaging* 37 (12) (2018) 2663–2674, <http://dx.doi.org/10.1109/tmi.2018.2845918>.
- [26] X. Xiao, S. Lian, Z. Luo, S. Li, Weighted res-UNet for high-quality retina vessel segmentation, in: 2018 9th Int. Conf. Inf. Technol. Med. Education (ITME), IEEE, 2018, pp. 327–331, <http://dx.doi.org/10.1109/itme.2018.00080>.
- [27] L. Jiao, L. Huo, C. Hu, P. Tang, Refined UNet: UNet-based refinement network for cloud and shadow precise segmentation, *Remote Sens.* 12 (12) (2020) 2001, <http://dx.doi.org/10.3390/rs12122001>.
- [28] R. Devillers, M. Nogue, A. Chan-Hon-Tong, G. Le Besnerais, J. Pichillou, Experimental analysis of aluminum-droplet combustion in solid-propellant conditions using deep learning, in: *Proc. 8th Eur. Conf. Aeronaut. Sp. Sci.*, 2019, <http://dx.doi.org/10.13009/EUCASS2019-593>.
- [29] V.S. Bochkov, L.Y. Kataeva, WUUNet: Advanced fully convolutional neural network for multiclass fire segmentation, *Symmetry* 13 (1) (2021) 98, <http://dx.doi.org/10.3390/sym13010098>.
- [30] H. Ma, Y. Zhang, O.J. Haidn, N. Thuerey, X. Hu, Supervised learning mixing characteristics of film cooling in a rocket combustor using convolutional neural networks, *Acta Astronaut.* 175 (2020) 11–18, <http://dx.doi.org/10.1016/j.actaastro.2020.05.021>.
- [31] B. Wang, B. Xie, J. Xuan, K. Jiao, AI-based optimization of PEM fuel cell catalyst layers for maximum power density via data-driven surrogate modeling, *Energy Convers. Manage.* 205 (2020) 112460, <http://dx.doi.org/10.1016/j.enconman.2019.112460>.
- [32] F. Perrotta, T. Parry, L.C. Neves, Application of machine learning for fuel consumption modelling of trucks, in: 2017 IEEE Int. Conf. Big Data, IEEE, 2017, pp. 3810–3815, <http://dx.doi.org/10.1109/bigdata.2017.8258382>.
- [33] F. Elmaz, B. Büyükcakır, O. Yücel, A.Y. Mutlu, Classification of solid fuels with machine learning, *Fuel* 266 (2020) 117066, <http://dx.doi.org/10.1016/j.fuel.2020.117066>.

- [34] A. Rüttgers, A. Petrarolo, M. Kobald, Clustering of paraffin-based hybrid rocket fuels combustion data, *Exp. Fluids* 61 (1) (2019) <http://dx.doi.org/10.1007/s00348-019-2837-8>.
- [35] T. Haas, C. Schubert, M. Eickhoff, H. Pfeifer, BubCNN: Bubble detection using Faster RCNN and shape regression network, *Chem. Eng. Sci.* 216 (2020) 115467, <http://dx.doi.org/10.1016/j.ces.2019.115467>.
- [36] J. Li, S. Shao, J. Hong, Machine learning shadowgraph for particle size and shape characterization, *Meas. Sci. Technol.* 32 (1) (2020) 015406, <http://dx.doi.org/10.1088/1361-6501/abae90>.
- [37] T. DeVries, G.W. Taylor, Leveraging uncertainty estimates for predicting segmentation quality, 2018, CoRR [arXiv:1807.00502](https://arxiv.org/abs/1807.00502).
- [38] MATLAB, Version 9.10.0.1613233 (R2021a), The Mathworks, Inc., Natick, Massachusetts, 2021.
- [39] N. Otsu, A threshold selection method from gray-level histograms, *IEEE Trans. Syst. Man, Cybern.* 9 (1) (1979) 62–66, <http://dx.doi.org/10.1109/tsmc.1979.4310076>.
- [40] ITU-R, Studio Encoding Parameters of Digital Television for Standard 4:3 and Wide-Screen 16:9 Aspect Ratios, (techreport Rec. ITU-R BT.601-7) International Telecommunication Union, 2011.
- [41] F. Chollet, et al., Keras, 2015, <https://keras.io>.
- [42] R. Chauhan, K.K. Ghanshala, R. Joshi, Convolutional neural network (CNN) for image detection and recognition, in: 2018 First Int. Conf. Secure Cyber Comput. Commun. (ICSCCC), IEEE, 2018, <http://dx.doi.org/10.1109/icsccc.2018.8703316>.
- [43] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, in: F. Bach, D. Blei (Eds.), *Proc. 32nd Int. Conf. Mach. Learning*, Vol. 37, PMLR, Lille, France, 2015, pp. 448–456, URL <http://proceedings.mlr.press/v37/ioffe15.html>.
- [44] M. Abdar, F. Pourpanah, S. Hussain, D. Rezazadegan, L. Liu, M. Ghavamzadeh, P. Fieguth, X. Cao, A. Khosravi, U.R. Acharya, V. Makarenkov, S. Nahavandi, A review of uncertainty quantification in deep learning: Techniques, applications and challenges, *Inf. Fusion* 76 (2021) 243–297, <http://dx.doi.org/10.1016/j.inffus.2021.05.008>.
- [45] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, *J. Mach. Learn. Res.* 15 (56) (2014) 1929–1958, URL: <http://jmlr.org/papers/v15/srivastava14a.html>.
- [46] T. Hastie, R. Tibshirani, J. Friedman, *The Elements of Statistical Learning*, Springer, New York, 2009, p. 309, <http://dx.doi.org/10.1007/978-0-387-84858-7>.
- [47] A. Buslaev, V.I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, A.A. Kalinin, Albumentations: fast and flexible image augmentations, *Information* 11 (2) (2020) 125.
- [48] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2017, [arXiv:1412.6980](https://arxiv.org/abs/1412.6980).
- [49] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, 2019, [arXiv:1711.05101](https://arxiv.org/abs/1711.05101).
- [50] N.S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, P.T.P. Tang, On large-batch training for deep learning: Generalization gap and sharp minima, 2016, CoRR [arXiv:1609.04836](https://arxiv.org/abs/1609.04836).
- [51] A. Weinstein, A. Gany, Testing and modeling liquefying fuel combustion in hybrid propulsion, in: *Prog. in Propuls. Phys.*, EDP Sciences, 2013, pp. 99–112, <http://dx.doi.org/10.1051/eucass/201304099>.
- [52] M. Karabeyoglu, B. Cantwell, D. Altman, Development and testing of paraffin-based hybrid rocket fuels, in: 37th Jt. Propuls. Conf. Exhib., AIAA, 2001, <http://dx.doi.org/10.2514/6.2001-4503>.
- [53] R.R. Picard, R.D. Cook, Cross-validation of regression models, *J. Amer. Statist. Assoc.* 79 (387) (1984) 575–583, <http://dx.doi.org/10.1080/01621459.1984.10478083>.
- [54] S. Arlot, A. Celisse, A survey of cross-validation procedures for model selection, *Stat. Surv.* 4 (2010) 40–79, <http://dx.doi.org/10.1214/09-SS054>.