

# AUTOMATIC DETECTION AND LOCALIZATION OF HYPERSONIC BOUNDARY LAYER STRUCTURES WITH DEEP TRANSFER LEARNING

Joseph Mockler<sup>1</sup>, Catherine Stock<sup>1</sup>, Chase Latyak<sup>1</sup>, Tyler Ostrowski<sup>1</sup>, Stuart Laurence<sup>1</sup>

<sup>1</sup>Dept. of Aerospace Engineering, University of Maryland, College Park, MD 20742

**Abstract.** Transition from laminar to turbulent flow in a hypersonic boundary layer significantly increases thermal loading and viscous drag, but the onset of transition is difficult to predict; experiments are often necessary to identify the mechanisms leading to turbulence. The dominant mechanism of turbulent development on slender bodies in hypersonic flow is trapped acoustic wave packets within the boundary layer, called second (or Mack) modes. Second-mode disturbances can be detected in schlieren images from hypersonic wind tunnel experiments, but identification is limited to visual inspection, prohibiting whole-video analyses with tens- or hundreds-of-thousands of images. Previous methods of automatically identifying second-mode wave packets have limited accuracy and are highly variable between experiments. Therefore, we propose using transfer learning, which includes using a pretrained convolutional neural network with a custom dense network, to automatically identify and localize the existence of second-mode instabilities and turbulence in schlieren video data. We implement our approach on six experiments from three different campaigns and draw comparisons with existing state-of-the-art identification methods. Finally, we demonstrate that the developed model shows promise to generalize to various experimental configurations, is applicable to tasks such as calculating wave packet propagation speeds, and shows improvement in accuracy compared to existing automated identification methods.

## 1. INTRODUCTION

(CUI) Turbulent flow in hypersonic boundary layers leads to enhanced surface thermal loads and a significant increase in drag. The growth and subsequent breakdown of boundary layer instabilities is the primary mechanism for turbulence development, but is often difficult to predict, leading to the overdesign of hypersonic vehicles. In hypersonic flows  $\geq$  Mach 5, the dominant instability mode on slender bodies is the second (Mack) mode, which is realized as trapped acoustic waves inside the boundary layer (Fedorov, 2011). The breakdown of these waves into turbulence is a continuous process, but is roughly divided into five stages: (i) laminar boundary layer flow, instabilities undergoing (ii) linear and (iii) nonlinear growth, (iv) a boundary layer with turbulent spots from previous instabilities, and (v) fully turbulent flow (Fedorov,

This content may contain elements of controlled unclassified information (CUI), unclassified, or information classified at a lower level than the overall classification displayed. This content shall not be used as a source of derivative classification; refer instead to Hypersonics for Military Systems and Applications Security Classification Guide, Feb 2021. It must be reviewed for both Classified National Security Information (CNSI) and CUI in accordance with DoDI5230.09 prior to public release.

DISTRIBUTION STATEMENT: C

WARNING -This document contains technical data whose export is restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751, et seq.) or the Export Control Reform Act of 2018 (Title 50, U.S.C., Chapter 58, Sec. 4801-4852). Violations of these export laws are subject to severe criminal penalties. Disseminate in accordance with provisions of DoD Directive 5230.25.

Controlled by:

Controlled by:

CUI Category:

Distribution Statement:

POC:

2011). In the linear growth phases, the second mode will take a wave number of roughly  $1/(2\delta_{99\%})$ , where  $\delta_{99\%}$  represents the 99% boundary layer height, and propagates at around 85% of the freestream velocity through the boundary layer. As growth becomes nonlinear, higher harmonics develop and eventually break down as this high frequency energy amplifies and grows into the chaotic, broadband frequency content that characterizes turbulent flow (Demetriades, 1974).

(CUI) The acoustic origin of these modes produces density variations in the boundary layer, resulting in refractive index gradients that may be imaged and visualized with schlieren photography. Recently, schlieren imaging has successfully supported quantitative studies of hypersonic boundary layer transitions by providing a nonintrusive method of measuring off-body flow disturbances (Casper *et al.* 2014, Laurence, *et al.* 2014, Laurence *et al.* 2016). This technique is successfully employed today to study shear-layer and second-mode boundary layer disturbances (Butler and Laurence 2022, Sousa *et al.* 2024). In schlieren images, second mode instabilities often appear as periodic, rope-like structures that spread from the surface towards the top of the boundary layer, and these prior experimental campaigns have captured second mode instability growth and breakdown to turbulence. Ranging from just a few milliseconds to seconds, schlieren video data from these experiments have produced hundreds of thousands of images. While the presence of such instabilities is clear in these images, the scale of imaging in the hypersonic regime prohibits meaningful manual identification. Moreover, advances in imaging technology, computation, and digital storage permit imaging at higher frequencies and the generation of significantly more experimental data is likely. Therefore, feature identification is a crucial task for global analysis of these experiments and presents the opportunity for significantly reducing required analysis time for researchers, aerospace designers, and engineers. As a result, automatic feature identification research has attracted attention in recent years. In this paper, we explore methods of identifying both linear, second-mode instabilities and fully turbulent flow to automatically characterize the breakdown process.

## 1.1 Existing Work

(CUI) First attempts at automated detection were made by Casper *et al.*, who proposed identifying second-mode instabilities with a correlation-based method (Casper *et al.* 2014). In their work, the researchers cross-correlated sections of an image twice the boundary height with an extended section of the image; second-mode packets were identified by peaks in such correlation downstream. The authors applied a Canny edge detection method, where strong contours of the image were calculated, to identify developed turbulence by flagging locations with edges above the boundary layer. Upon analysis with pressure measurements, the authors reported agreement with schlieren video data, but note additional techniques were needed for improved identification. Subsequently, Shumway proposed additional methods for second-mode and turbulence identification, compared it against Casper's work, and extended the test set to identify experimental conditions where the methods may be most applicable (Shumway *et al.* 2015). In this work, the researchers proposed (i) filtered image, (ii) short-time Fourier transform (STFT), (iii) MUSIC, and (iv) wavelet methods for second-mode identification and (i) modified wavelet, and (ii) high frequency methods for turbulence identification.

(CUI) These four methods of second-mode identification are driven by a frequency-based detection analysis. Starting from a background-subtracted and bandpass filtered image, filtered image moves downstream and identifies candidate wave packet locations by checking for increasing and decreasing maxima spaced by the second-mode wavelength,  $\lambda_{2nd}$ . STFT extends this by calculating the power spectrum of a

windowed region downstream and determining the ratio between the power around the expected wave packet frequency over the low-frequency power spectrum; high ratios indicate likely wave packets. The MUSIC, or Multiple Signal Classification, technique estimates the pseudospectrum with a correlation matrix of the frame (Cheney, 2001). Similar to STFT, a windowing method is shifted downstream and the pseudospectral information is used to identify likely second-mode wave packets. The wavelet method makes use of the wavelet function, which can be both shifted for localization and dilated for wavelength information (Farge, 1992). This function is convolved downstream to provide both phase and magnitude information; large magnitude response then identifies a wave packet and an estimate of location. To detect developed turbulence, authors turn again to frequency-based methods. The high frequency method again uses an STFT to determine regions with the high frequency content characteristic of developed turbulence. The modified wavelet method detects turbulence by noting that developed turbulence has high frequency content around the first harmonic of the second-mode frequency and searching for such with a wavelet function.

(CUI) These methods require a pre-defined threshold to extract meaningful results, and this can be a challenging process, where ill-tuned parameters may yield poor results (Shumway *et al.* 2015). Moreover, the authors report that a method's accuracy varies strongly with the experimental set it's tested on, with some methods finding virtually no wave packets in a set of 4,000 images and others finding over 2,000 wave packets. A primary challenge lies in developing a method that is experiment-agnostic and can be successful in a range of hypersonic flow and schlieren visualization conditions. Therefore, our work focuses on assessing new image classification technologies to address this limitation.

(CUI) Convolutional Neural Networks (CNNs) have found enormous success in traditional data mining environments for image classification (LeCun *et al.* 2015) by optimally mapping high-dimensional data (e.g. image data) to a lower dimension or single classification. Significant fundamental research into CNNs dates back to the 1990s, where researchers found that CNNs, which are connected networks that map regions of an image to simpler representations using nonlinear functions, could extract elementary features of the image such as oriented edges, end-points, and corners (LeCun *et al.* 1995). When coupled with additional pooling layers (which perform grouped operations over this mapped representation), researchers discovered that these mappings may be optimized to separate, and thus classify, differing images. The process of determining such an optimal mapping is known as supervised learning, where a gradient-based optimization routine compares the ground-truth (provided by a human) with the image's latent representation to determine a classifier's CNN parameters. Often, this is done in smaller, randomly sampled batches to stochastically optimize the CNN parameters, avoiding local minima in the high-dimensional, nonlinear parameter space (LeCun *et al.* 2015). The first major breakthrough in CNN development was AlexNet, a CNN capable of classifying over 1,000 different types of images (Krizhevsky *et al.* 2012). This work promoted the development of hundreds of other image classifiers in the decade since, using similar CNN architectures (Weiss *et al.* 2016).

(CUI) These architectures have profound classification performance, but are enormous in scale. AlexNet, for example, required 1.2 million labeled images to train its network of 60 million parameters, requiring stacks of parallelized GPUs to perform the training (Krizhevsky *et al.* 2012). For smaller image classification problems, this represents a significant burden, and thus researchers have adopted transfer learning as a solution. Transfer learning uses an existing trained architecture, such as AlexNet, and adds on a single-layer dense neural network to perform domain-specific image classification. Therefore, researchers

must only train the dense, top layer of this network, requiring significantly fewer images and much lower computational cost (Weiss *et al.* 2016). One early experiment, for example, successfully used ImageNet2012, a pretrained CNN, to identify an entirely different image set using only a small top layer network (Oquab *et al.* 2014), showcasing the scale and ease of this approach. Since then, various communities have adopted transfer learning to tackle domain-specific problems, like medical image classification (Wang *et al.* 2024, Kim *et al.* 2022), hyperspectral satellite imaging (Lin *et al.* 2018), and crowd analysis (Alrowais *et al.* 2022). The trade-off in lower computational cost and fewer training images is that the classifier becomes domain-specific and is quite sensitive to the images it is trained on. For hypersonic boundary layer analysis, this architecture presents an opportunity to successfully identify experiment-specific structural flow features. By labeling and training on just a few schlieren images (around 150), transfer learning shows promise for training and classifying boundary layer features on an individual researcher's machine without the burden of developing complicated CNN architectures. Therefore, our work focuses on using a transfer learning approach to identify key structural features in hypersonic flow.

## 1.2 Scope and Contributions

(CUI) In this paper, we propose using a transfer learning approach to classify hypersonic flow features, namely second-mode linear wave packets and fully developed turbulence. Specifically, we begin with a pretrained convolutional neural network, ResNet50 (He *et al.* 2016). This network handles broad feature extraction, but replacing the pretrained classification layer with a custom dense layer allows the model to be tuned specifically to the hypersonic schlieren image domain. Under this transfer learning approach, we gain the ability to classify massive sets of data from a model trained only on 100-200 manually labeled images, a significant reduction in workload for researchers to analyze structures and instabilities in full schlieren videos.

(CUI) We also compare the results of machine learning (ML) techniques to those of existing methods, including MUSIC, auto-correlation, filtered image, STFT, and wavelet for second-mode instabilities, and STFT and Canny for turbulence. We foremost showcase the functionality and novelty of our approach through significantly improved global accuracies. In addition, our approach is much more consistent in performance across experimental conditions, marking an improvement over the extreme variability of existing automated methods. We illustrate the broad capabilities of our approach by using the same network structure to produce separate models that can identify either second-mode waves or turbulence, both to a high degree of accuracy. We also demonstrate the ability to incorporate these models into other existing hypersonic analysis routines, to perform tasks like wave packet propagation speed or intermittency calculations. The combination of accuracy, consistency, and versatility that we illustrate in this paper indicate the value and novelty of applying transfer learning techniques to the field of hypersonic analysis.

(CUI) Our paper is structured as follows: we begin by detailing the general approach to creating and deploying a machine learning model in Section 2. In Section 3, we discuss in detail the performance of our classifier on second-mode wave packets, followed by performance on turbulence. We also detail additional performance metrics that were employed to better understand how the model performs, especially with highly skewed data sets. We directly compare the performance of our method to existing techniques for each experiment, showcasing the efficacy of machine learning in this context. Section 4 integrates our approach with existing hypersonic boundary layer analyses to showcase its potential applications. This section also discusses the pursuit of a global classification model that applies to any schlieren video, regardless of

Mach or Reynolds number, and pursues an understanding of the classification process. Finally, in Section 5, we conclude the paper and propose areas for future work to continue exploring the topic.

## 2. APPROACH

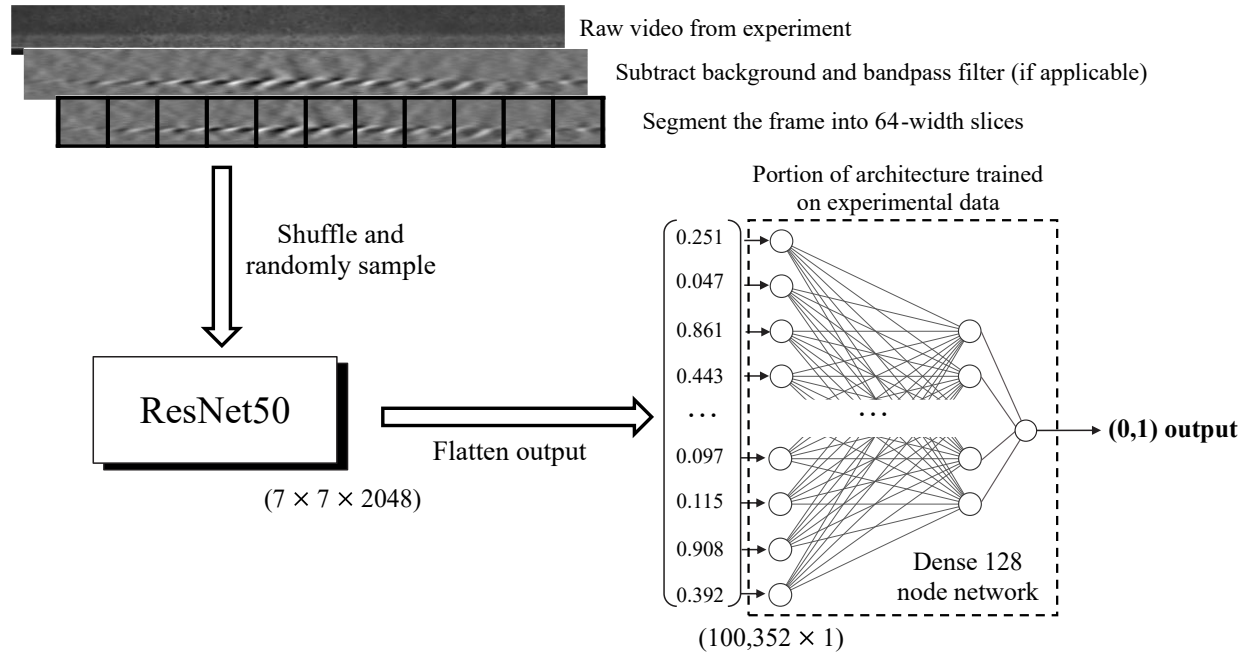
(CUI) Significant work in hypersonic transitional work concerns the development and subsequent breakdown of second-mode instabilities into turbulent flow. The automatic identification of these features is crucial for understanding the mechanisms of turbulent development among large datasets (e.g. schlieren flow videos) and is often a first step in more sophisticated boundary layer video analyses (Shumway *et al.* 2015). Therefore, we propose an ML-based approach for classification and localization of such features and divide this section into two subsections. We first discuss building the classifier via a transfer learning neural network architecture, then we discuss how to deploy the classifier model on a large data set for whole-experiment classification. Combined, this training-and-deployment approach may be applied to a sweeping range of hypersonic flow conditions, which we demonstrate in Section 3.

### 2.1. Classifier Training

(CUI) Advances in machine learning, data mining, and faster computing have introduced image classification architectures, often based on convolutional neural networks, capable of identifying and localizing subjects in real-world images (Weiss *et al.* 2016). Among these advancements is transfer learning, which is an image processing technique for converting a general deep image classification architecture into application-specific object detection model, often requiring orders of magnitude fewer images to develop a successful detection model (Weiss *et al.* 2016). The distinction between transfer learning and more traditional CNN architectures lies in which parameters are optimized for classification: transfer learning uses a pre-existing CNN to extract features from an image, which are fed to a dense neural network with single sigmoid activation function top layer. The parameters of this dense neural network are optimized during training while the pre-existing CNN is kept fixed, and thus, this hybrid architecture becomes problem-specific classifier. We leverage this specificity to build individual classifiers that detect and distinguish second-mode wave packets or turbulence.

(CUI) Training requires ground-truth labeled samples to optimally determine parameters that map extracted features to correct classification. Therefore, the first step is data gathering and pre-processing. To train the dense neural network for second-mode wave packet or turbulence classification, we first background-subtract the mean frame from an experimental video, where the mean is taken every 100 frames. When building a second-mode classifier, we then bandpass filter the frames around the expected wavelength using a standard Butterworth filter. The cutoff frequencies are automatically selected based on the boundary layer thickness ( $\delta$ ) to filter around the frequency of the second mode, usually set to  $1/4\delta$  and  $1/\delta$  to enhance the wave packet's appearance. When building the turbulence model, no filtering is required. Then, we manually label frames of interest with an approximate boundary box around the structures we seek to detect, until a desired number of labeled frames is met. Finally, most pretrained CNNs, including ResNet50, require square image inputs. Because boundary layer images are inherently high aspect (e.g. 20:1), reshaping entire frames to squares adversely distorts flow structures. Therefore, we slice the image into approximate squares which then train the dense neural network classifier. When rescaled to meet pretrained network requirements, the flow structures are better preserved and are clearly visible. Quite advantageously, slicing provides a degree of localization because the binary present-or-not results are now defined on a slice level. For example, a 1280-pixel wide video may be divided into 20 64-pixel slices structure may be localized within

less than 64-pixels, or 5% of the frame, providing good approximate localization (examples of this are provided in later sections). We detail this entire procedure in Figure 1.

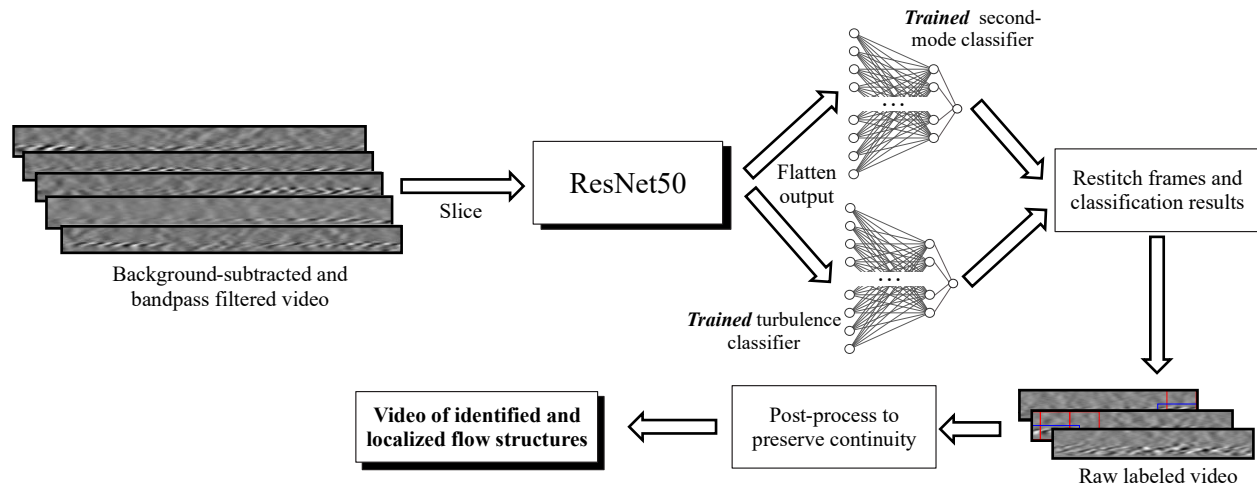


**(CUI) Figure 1: Proposed transfer learning architecture and workflow for training and identifying key hypersonic boundary layer flow features, including second-mode instabilities and developed turbulence.**

(CUI) The image slices and their ground-truth binary label now train the dense neural network. Features are first extracted by passing the rescaled image slices through the pretrained CNN. We use ResNet50 due to its reported success in the medical imaging community, where images also primarily represent variations in density and have a similar composition to schlieren images (Kim *et al.* 2022). However, note that this approach is not limited to any specific pretrained CNN. The dense network optimization is performed to train the second-mode and turbulence models, one unique network for each. We use a 256 or 128-node dense network with ReLu activation,  $1e-4$   $l_1$  and  $l_2$  kernel regularization,  $1e-4$  bias regularization,  $1e-5$  or  $1e-6$  learning rate, 50% dropout, and a final sigmoid activation function for improved mapping to a binary. Variations in total nodes in the dense layer and learning rate were found to have virtually no impact on overall accuracy and model performance, and were simply changed based on the computing capabilities of different personal machines used for training. See the Appendix for detailed training accuracy and loss graphs for each experiment. Regularizations and dropout were initially tuned to improve model robustness and generalization within an experiment. Our model is optimized with a binary cross entropy loss function, a batch size of 16, a 25% train/validation data split, and run for 20 epochs until the loss converged or validation accuracy steadied. Training and neural networks were implemented with Keras, a popular Python machine learning toolbox (Chollet 2015). We note that classification models are experiment-specific, so a second-mode classifier was built for each experiment, and a turbulence classifier was built for each high-turbulence experiment (see Table 1, Section 3.1). Finally, training is performed on only a small subset of images, limited by the number of frames in an experiment that must be labeled with a ground truth. While labeling may be a time-consuming process, the advantage of a well-performing classifier is deployment at scale by classifying an entire video (which may have tens of thousands of images) to identify and localize flow structures.

## 2.2. Classifier Deployment

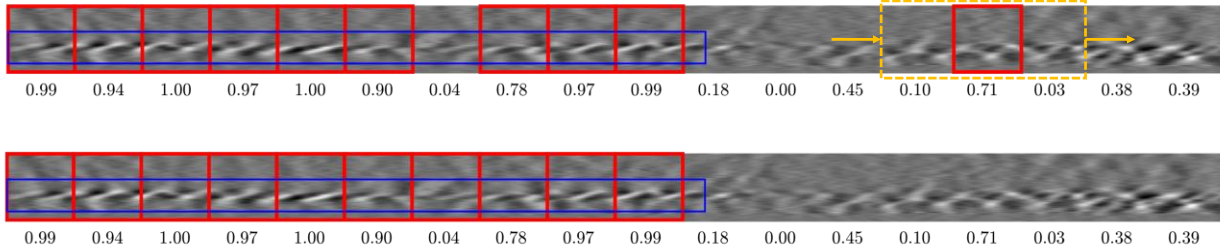
(CUI) In this subsection, we overview classifier deployment on an entire experimental video. As in training, the frames of an experimental schlieren video must be first background subtracted, normalized, and bandpass filtered (if applicable). Then, the frames are sliced, and features are extracted by passing the images through ResNet50. Finally, these features are passed to both second-mode and turbulence classifiers and mapped to a (0,1) value, quantifying the presence of a flow structure in the slice. As an initial result, slices may be classified with a present-or-not binary using simple rounding. This classification provides a reasonable performance, but is improved with post-processing. This process is graphically presented in Figure 2.



(CUI) **Figure 2: Architecture and workflow for using a trained transfer learning classifier to identify and localize hypersonic boundary layer features from a schlieren video.**

(CUI) This classifier produces a few spurious false positives, where a flow feature is labeled, but is not actually present, and spurious false negatives, where a flow feature is present, but not labeled. However, knowing the flow structures tend to be much wider than an individual slice, we may compensate for such incorrect results by considering physical connectivity between the classified slices. We introduce a windowing routine to reduce spurious results by taking a moving average of classification along the slices and setting the local classification threshold to be based on this value. Additionally, for small features, if an individual slice has a particularly high classification value, this is relabeled as a breakout and overrides the windowing results. We plot an example of our processing technique in Figure 3.

(CUI) The raw classification results from Langley Re7.7 (detailed in Section 3) are plotted above with two incorrect labels: one false positive downstream and one false negative upstream. Using a windowing technique considers the local cohesion of the flow structure, and in this case, eliminates both false negative and false positive results, returning a completely accurate labeling of the frame. After, if a slice is labeled with both turbulence and a second-mode wave packet, the label with a higher classification value overrides the windowing result. Our approach is then tested on a broader range of experimental conditions to assess performance and range of application.



(CUI) **Figure 3: An example of raw classification (top) and the windowed classification (bottom), with raw (0,1) classification values printed below each slice. The top image shows a false positive and negative, with the yellow box representing a moving 3-slice window that filters out the false positive because the local sum (0.84) is less than the pre-defined threshold of 1.5. The red boxes represent the classifier results, while the blue boxes are the human-labeled ground truth.**

### 3. CLASSIFICATION PERFORMANCE

(CUI) This section demonstrates the flexibility and broad application of our approach by classifying second-mode wave packets in three diverse experimental campaigns with six total video experiments and turbulence in three of those video experiments. We first present an overview of the experimental conditions we seek to classify, then summarize the performance results.

#### 3.1 Experimental Overview

(CUI) We gathered three experimental campaigns of schlieren images of high-enthalpy hypersonic flow over conical models to test our approach on. Run 4120 from the AEDC Tunnel 9 experimental campaign was conducted in a blowdown hypersonic wind tunnel using a 1.55 m long 7° half-angle cone. The data set described in Table 1 captures the top surface of the cone at a 0° angle of attack, although the full experiment involved sweeping through increasing angles to study transition to turbulence. The Cone Flare data sets were collected in the University of Maryland’s HyperTERP, a Mach 6 reflected shock tunnel. These experiments used a 410 mm long, 5° half-angle cone to examine boundary layer transition resulting from a sudden surface angle increase, but our videos only identify structures at the initial 0° angle of attack. The Langley data sets came from the 20-inch Mach 6 Wind Tunnel at NASA Langley. These runs investigated the nonlinear development of second-mode wave packets over a 0.52 m long, 7° half-angle cone. Additional details on experimental set-up and methods may be found in (Kennedy *et al.* 2018), (Butler and Laurence, 2022), and (Sousa *et al.* 2024) for the Tunnel 9, Cone Flare, and Langley campaigns, respectively. These tests represent a broad range of Reynold’s numbers, cone geometries, and freestream velocities common to hypersonic transitional boundary layer experiments.

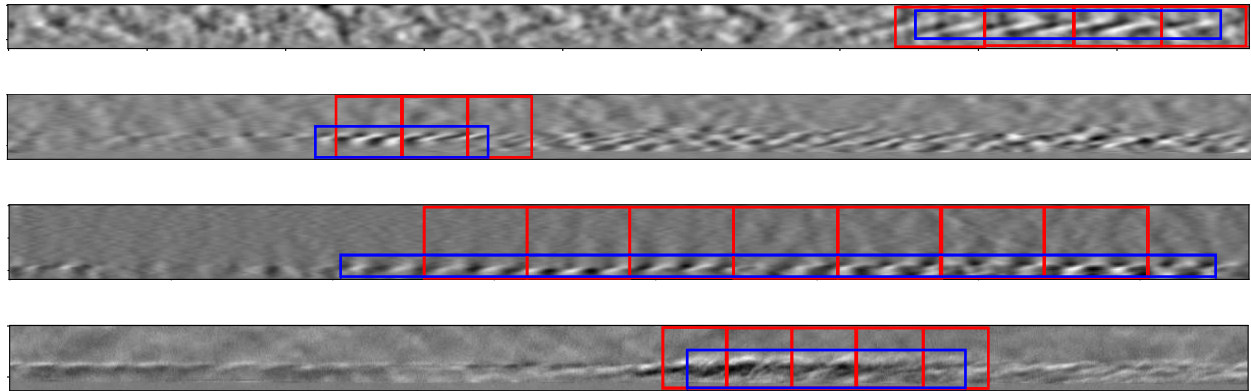
(CUI) **Table 1: Summary of experimental video where turbulence and second-mode instabilities were classified and localized. Check marks indicate which classification models were built and tested.**

Exp. Campaign	Test Conditions	Re $\times 10^6$ , 1/m	Image Size	Labeled Frames	2 <sup>nd</sup> Mode	Turb.
Tunnel 9	7° Cone, Mach 14	3.6	748 x 48	400	✓	
Cone Flare	5° Cone, Mach 6	3.3	896 x 32	500	✓	
Cone Flare	5° Cone, Mach 6	4.5	640 x 32	500	✓	✓
Langley	7° Cone, Mach 6	6.4	1280 x 64	500	✓	
Langley	7° Cone, Mach 6	7.7	1280 x 64	500	✓	✓
Langley	7° Cone, Mach 6	9.7	1280 x 64	500	✓	✓



### 3.2 Initial Classification Performance

(CUI) In each experiment, only 150 to 200 labeled frames were used to train the wave packet models, with merely 100 images used to train the turbulence models; the remaining labeled images were used to test the classification performance. Some representative classification results are shown in Figure 4. In each, the bulk of the flow structure is successfully detected by the classifier with positive slices detecting the main presence of the structure. Further, the structure-specific classifiers distinguish between second-mode instability and turbulence well. The Langley Re7.7 second-mode model (Figure 4), for example, detects just the upstream second-mode instability without overclassifying the turbulence downstream. Similarly, the Langley Re9.7 turbulence classifier successfully detects the developed turbulence downstream, but does not overclassify the second-mode instabilities upstream in the same frame (Figure 4). Finally, the connected positive identifications provide a reasonable estimate of the location of the flow structure, which we will exploit in later sections for more advanced boundary layer analyses.



(CUI) Figure 4: Example classified second-mode wave packet images from Cone Flare Re4.5, Langley Re7.7, and Tunnel 9, from top-to-bottom respectively. The bottom-most image is an example of classified developed turbulence from Langley Re9.7.

(CUI) Table 2: Summary classification performance results in labeling boundary layer flow structures

Experiment, Re	Structural Detection	% True Positive	% True Negative	% False Positive	% False Negative	Accuracy
Tunnel 9, 3.6	Second-mode	21.9%	68.7%	2.77%	6.69%	90.5%
Cone Flare, 3.3	Second-mode	1.98%	95.1%	0.47%	2.48%	97.0%
Cone Flare, 4.5	Second-mode	4.61%	89.5%	0.84%	5.10%	94.1%
Langley, 6.4	Second-mode	10.1%	88.1%	0.55%	1.29%	98.1%
Langley, 7.7	Second-mode	31.7%	66.3%	2.57%	4.37%	93.1%
Langley, 9.7	Second-mode	25.7%	66.4%	2.32%	5.60%	92.1%
Cone Flare, 4.5	Turbulence	22.1%	74.9%	1.44%	1.56%	97.0%
Langley 7.7	Turbulence	6.97%	91.8%	0.45%	0.72%	98.8%
Langley, 9.7	Turbulence	3.92%	95.1%	0.11%	0.91%	98.9%

(CUI) Visual inspection provides a general understanding of classification performance, but quantitative measures will provide a more detailed evaluation. These flow structures represent continuous evolution from instability to turbulence, and thus, their classification, whether human or via neural network, is subject

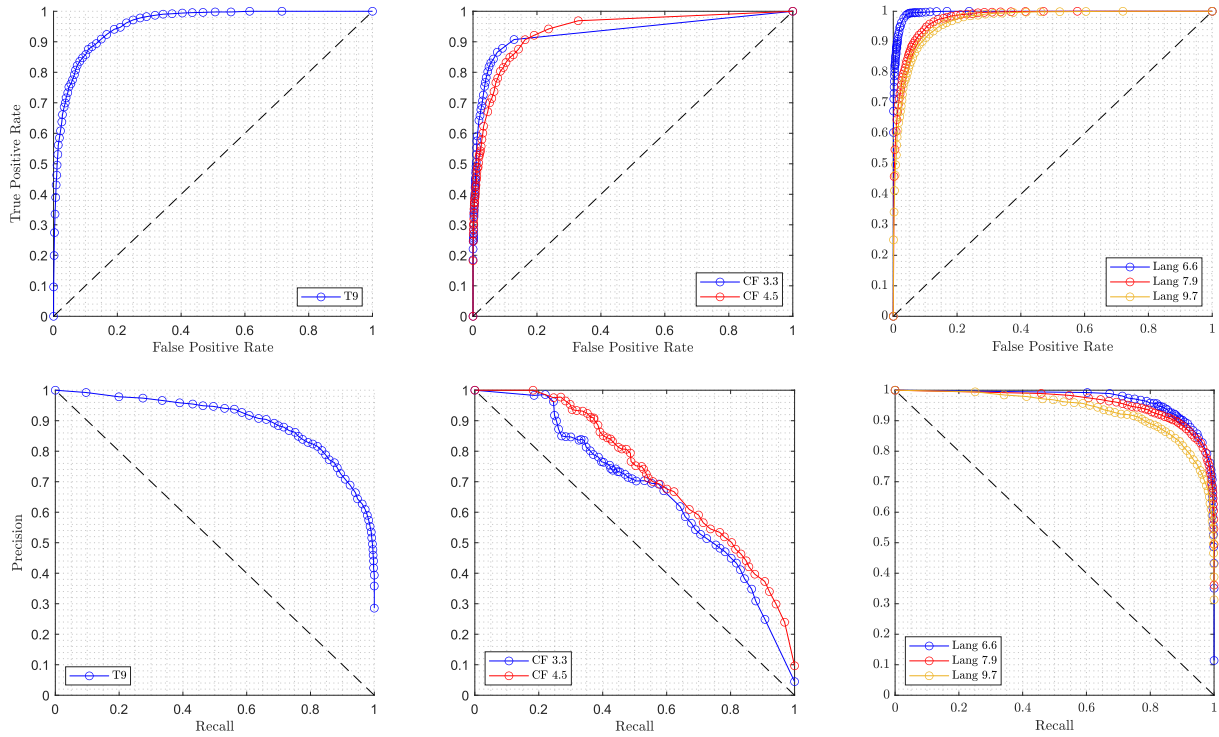
to some uncertainty. Any performance metric is influenced by this subjectivity, but evaluation over a sufficient number of images should provide a reasonable portrait of the classifier's performance. We assess the performance of our proposed approach by considering the rate of accurate classifications, false positives, and false negatives, all found in Table 2. As well, we present receiver-operating characteristic and precision-recall curves as evidence of classification success in the structure-specific sections that follow.

(CUI) Second-mode classification accuracy was above 90% in all experimental cases, and turbulence boasted 97% or above in all tested cases. The false positive and negatives rates were much smaller than their true counterparts, but the false negatives tended to be larger than the false positives across the tested experiments, indicating that the classifier is slightly conservative in its identification. Some of the tested experiments were quite sparse in features to detect; Cone Flare saw very few second-mode structures, and Langley Re9.7 was limited in turbulence. High accuracy could be driven by the scarcity of flow features of these sets – if a classifier were to simply label everything as negative, the resulting accuracy would be quite high from there being few flow structures anyways. In short, accuracy is a biased performance metric in imbalanced, or skewed, data sets. Therefore, we explore more robust metrics for characterizing classifier performance in the next two subsections.

### 3.3. Second-Mode Instability Classification

(CUI) Because flow, particularly at low Reynolds numbers, may only sparsely contain second-mode instabilities, robustly assessing classifier performance is crucial for understanding how it may perform on imbalanced sets. In some of our experiments, we already see this. Cone Flare data sets only contained 5-10% second-mode instabilities, and while their reported accuracy is quite high, we must separate the positive and negative classification effects to understand how well our approach classifies infrequent second-mode packets. One common method is to plot receiver-operating characteristic (ROC) and precision-recall (PR) curves (Davis *et al.* 2006). In short, an ROC curve is generated by varying the classification threshold then plotting how the number of correctly classified positive examples varies (typically increasing) against the number of incorrectly classified negative examples (typically decreasing). A PR curve, contrastingly, plots how precision, a measure of false positive fractions, and recall, a measure of false negative fractions, change with an increasing classification threshold, and is considered more robust to sparse data sets (Davis *et al.* 2006). We first create and plot ROC and PR curves for these classifiers, shown below.

(CUI) For a perfectly performing classifier, the ROC curve should tend towards the upper left of the plot, while the PR curve should tend towards the upper right. A unit positive slope through the intercept of the ROC curve indicates classification performance of equal guessing, and therefore, successful classifiers will be well above this line. A unit negative slope through the PR curve indicates guessing with knowledge of the underlying distribution, and therefore, the distance of a plotted PR curve from this unit line indicates performance as robust to skewness.



**(CUI) Figure 5: Second-mode instability classification performance among the six tested experimental runs with the top rows plotting the classifier’s ROC and the bottom rows plotting the PRC.**

(CUI) In all three experimental campaigns, the ROC curve tends towards the upper left, demonstrating strong classification performance on more balanced data sets. In Tunnel 9 and Langley campaigns, the PR curve tends strongly to the upper right, indicating good classification performance on skewed data sets. In Cone Flare experiments, the PR curve is above the unit slope line, but not as significantly as the other sets. The imaging signal-to-noise of this set is much lower than the more successful sets (see Figure 4 for visualization), making distinction between second-mode structures and noise more difficult than the other experimental sets. This classification performance measure, however, is relative to guessing with knowledge of the underlying distribution, which is not necessarily known. These classifiers, therefore, are successfully detecting second-mode instabilities in both even and unevenly distributed data sets.

(CUI) Finally, we compare our approach to five state-of-the-art second-mode detection algorithms and two turbulence detection algorithms. These algorithms generally rely on automatically computing and interpreting frequency content within the image, but vary in the approach for computing and extracting such limits. As such, our algorithm reasons only over nonlinear relationship between pixels inside a slice, differing in its fundamental approach compared with these methods. Our implementations of the comparison algorithms closely follow (Shumway *et al.* 2015), and we encourage the reader to review Shumway’s work for a complete description of these methods. The table below compares second-mode identification performance across methods.

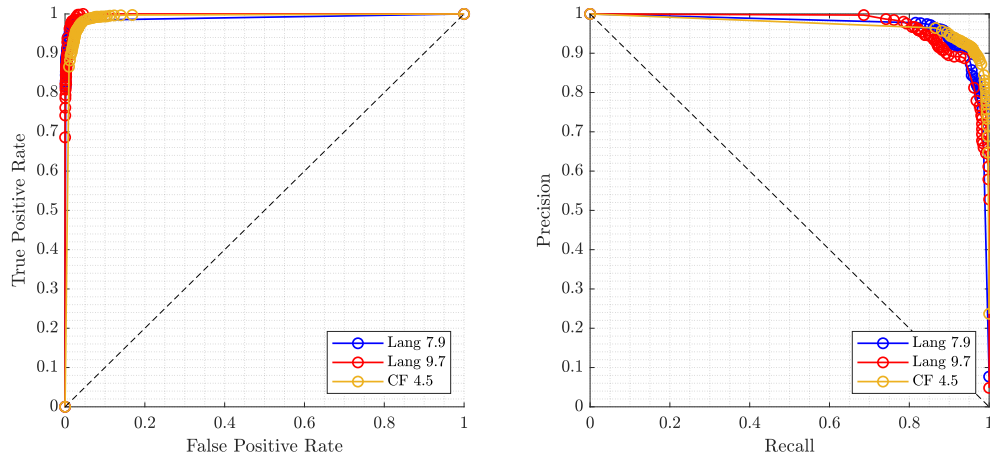
**(CUI) Table 3: Summary second-mode classification performance of our approach compared with existing state-of-the-art methods. The first number is accuracy, while the second and third below it are precision and recall, respectively.**

Experiment, Re	Our Approach	MUSIC	Auto-Correlation	Filtered Image	STFT	Wavelet Analysis
Tunnel 9, 3.6	90.5% (89%, 77%)	70.5% (27%, 84%)	76.8% (54%, 49%)	76.1% (54%, 27%)	76.8% (52%, 84%)	81.3% (60%, 77%)
Cone Flare, 3.3	97.0% (81%, 44%)	95.1% (0%, 0%)	95.6% (0%, 0%)	96.3% (0%, 0%)	74.9% (6%, 44%)	84.6% (2%, 7%)
Cone Flare, 4.5	94.1% (84%, 47%)	78.1% (7%, 16%)	92.1% (5.0%, 0.3%)	89.5% (5.5%, 2.4%)	90.8% (29%, 14%)	84.6% (8%, 10%)
Langley, 6.4	98.1% (95%, 89%)	92.5% (68%, 52%)	91.4% (66%, 36%)	92.8% (83%, 38%)	91.3% (66%, 34%)	93.7% (92%, 44%)
Langley, 7.7	93.1% (93%, 88%)	73.0% (57%, 90%)	79.9% (73%, 65%)	82.7% (77%, 71%)	68.6% (53%, 84%)	75.3% (59%, 89%)
Langley, 9.7	92.1% (92%, 82%)	62.9% (42%, 72%)	75.8% (67%, 40%)	79.7% (67%, 61%)	71.6% (31%, 77%)	67.1% (36%, 82%)

(CUI) In all experimental cases, our approach demonstrated the highest accuracy, highest precision, and in most cases, the highest recall. To start, our method produced higher precision, a measure of the false positive rate, than recall, a measure of the false negative rate, indicating again that our classifier is slightly conservative in its classification. In high signal-to-noise experimental campaigns (Langley and Tunnel 9 sets), the existing methods successfully detected some flow structures, but performed worse than our classifier in nearly all metrics. In sparse, high signal-to-noise sets (such as Langley Re6.4), the existing methods produced low recall as a result of under-classification. In the low signal-to-noise Cone Flare set, the existing methods failed to classify virtually any second-mode structures successfully, often producing precision and recall values near 0%, while our method was able to correctly identify most of the wave packets. Finally, between experimental campaigns, the existing methods were highly inconsistent, with accuracies varying by over 20% between experimental sets. Our method, however, is consistently accurate, and indicates much broader applicability to various experimental and imaging conditions compared with existing methods.

### 3.4. Turbulent Flow Classification

(CUI) The classification metrics described in Section 3.3 are deployed here to quantify the performance of our model on turbulence classification. ROCs and PRCs help to illustrate how successful the model is performing even in the case of highly skewed data, which is a common trend in labeling turbulence in the aforementioned data sets. In fact, only the Cone Flare Re4.5, Langley Re7.7, and Langley Re9.7 experiments contained enough turbulence to produce any meaningful results. The ROC and PR curves for these three data sets can be seen below in Figure 6. It is evident that in all three cases, the ROC curves are almost exactly in the upper left corner, and similarly the PR curves are pulled very high to the top right corner. These graphs show, along with the data provided in Table 4, that the turbulence classifier performs extremely well even in low turbulence, highly skewed sets.



**(CUI) Figure 6: Developed turbulence classification performance among the three tested experimental runs with significant turbulence present. Left plots the classifier’s ROC curves and right plots the PR curves.**

(CUI) As done with the classification of second-mode instabilities, we compared our classifier’s turbulence performance to existing techniques, namely STFT and Canny. For more detail on these algorithms, the reader is again referred to (Shumway *et al.* 2015). The performance comparison between our approach, STFT, and Canny is shown below in Table 4. Our approach achieved the highest accuracy, precision, and recall in every case. While the other methods obtained fairly high accuracies in both of the Langley sets, the precision and recall values are significantly lower than what our ML approach got. These sets contained much less overall turbulence (both under 8%) than the Cone Flare Re4.5 (24%), but the high values for both precision and recall in our approach indicates that it was able to successfully classify the turbulence that was present. On the other hand, the more balanced data set of Cone Flare Re4.5 resulted in the largest difference in accuracy between our method and the better of the other two, with an improvement of 8.4%. Once again, the precision and recall values for our approach are both noticeably higher as well, revealing that this technique performs well regardless of the percentage of features present in the data.

**(CUI) Table 4: Summary turbulence classification performance of our approach compared with existing state-of-the-art methods. The first number is accuracy, while the second and third below it are precision and recall, respectively.**

Experiment, Re	Our Approach	STFT	Canny
Cone Flare, 4.5	97.0%	88.6%	64.4%
	(94%, 93%)	(85%, 62%)	(31%, 45%)
Langley, 7.7	98.8%	84.5%	92.1%
	(94%, 91%)	(21%, 39%)	(44%, 25%)
Langley, 9.7	99.0%	97.2%	88.3%
	(97%, 81%)	(68%, 71%)	(17%, 43%)

(CUI) It is visually and quantitatively evident that our approach can consistently classify both second-mode wave packets and turbulence within hypersonic boundary layer images. The ability to train a fully capable model on such a small set of images allows for large-scale automated detection of these flow features in full schlieren videos. This enables us to apply our model to perform additional hypersonic analysis beyond simple structure detection, and such applications are discussed in Section 4.

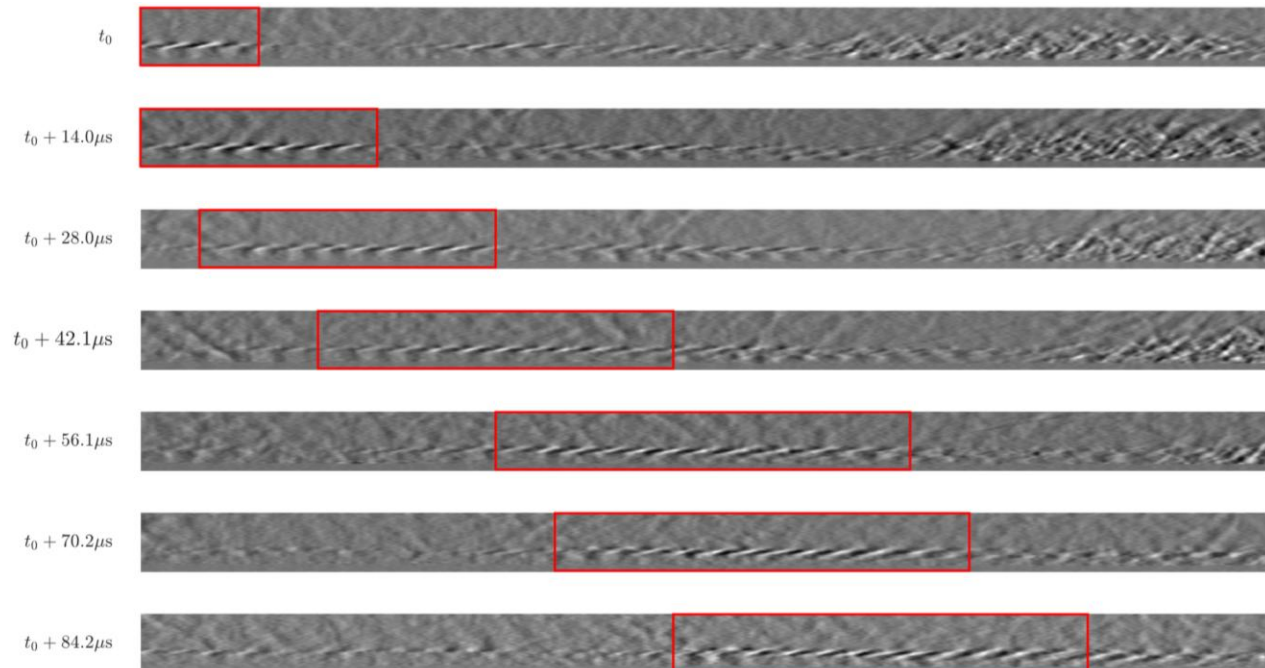
#### 4. APPLICATION TO BOUNDARY LAYER ANALYSIS

(CUI) Automated analysis is crucial for studying and quantifying large schlieren data sets that would otherwise be impossible to study to their full scale. In this section, we present a few applications for our automated analysis to showcase how our approach may fit into existing hypersonic boundary layer techniques and promote deeper analysis of large experimental videos.

##### 4.1. Propagation Speed Measurements

(CUI) Second-mode instabilities are the dominant instabilities leading to transition on slender, axisymmetric bodies above Mach 5 (Federov, 2011). Thus, studying their propagation and growth is the foundation of characterizing the transition to turbulence on aerospace vehicles. Moreover, correct characterization of second-mode instabilities is crucial for more sophisticated frequency-based analysis and necessary for inter-frame signal reconstruction analyses (Sousa *et al.* 2024). In this subsection, we showcase how our classification approach may be used to compute the second-mode propagation speed inside the boundary layer.

(CUI) We first train a classifier, as detailed in Section 2.1, to classify second-mode instabilities on each of the Langley experiments. Then, we deploy the classifier on a 10 ms section of steady-state hypersonic flow with 2,500 frames, as detailed in Section 2.2, for each of the three experiments; the flow and imaging details may be found in (Sousa *et al.* 2024). The classification returns approximate second-mode locations in each of the frames (if present) that show the propagation of a second-mode instability along the boundary layer. Knowing the approximate location of advecting instabilities and the time between frames, the correlation between located second-mode instabilities can be used to determine the propagation speed between frames; this correlational analysis is a popular approach for measuring propagation speeds, and details may be found in the following resources: (Laurence *et al.* 2014, Sousa *et al.* 2024, Kennedy *et al.* 2018). We present one visual example of a successfully identified instability as it propagates through the boundary layer.



**(CUI) Figure 7: An identified and tracked second-mode instability propagating downstream from the Langley Re7.7 experimental video.**

(CUI) To calculate the second-mode wave packet propagation speed, we filter the identifications for sufficient length, a minimum of 128 pixels long, and for at least three frames of sequential identifications to avoid false positives. Then, among the set of frames with instabilities present, the propagation speed is calculated using a correlation analysis between subsequent frames. The correlation peak within a range of expected propagation speeds (500 – 1,000 m/s) is determined and its associated lead is used to calculate the between-frame propagation speed. Our results, compared with the results when using MUSIC to detect and localize second-mode wave packets (Sousa *et al.* 2024), are found in Table 5.

**(CUI) Table 5: A comparison of propagation speeds with second-mode instability locations determined by our method vs existing methods. Reproduced from (Sousa *et al.* 2024)**

Experiment, Re	Up, m/s, ours	Up, m/s, MUSIC	$\pm\sigma$ , m/s, ours	$\pm\sigma$ , m/s, MUSIC
Langley, 6.4	828.5	835.3	36.9	40.6
Langley, 7.7	814.6	824.6	11.5	20.9
Langley, 9.7	818.1	819.2	13.2	11.5

(CUI) Propagation speeds using our method agree with those presented in (Sousa *et al.* 2024) and show a slightly improved measurement error in two of three. While it is difficult to necessarily assign improved measurement error to using our classification method over MUSIC, our analysis demonstrates the possibility of easily integrating our classification scheme into existing propagation analysis techniques. Coupled with results indicating that our approach is much more consistent, integration into propagation speed calculations would be applicable to a wider range of experiments than MUSIC may be. We extend this in the next section to show how our classification approach may be combined to identify sections of a video where propagating instabilities breakdown into turbulence within a field of view.

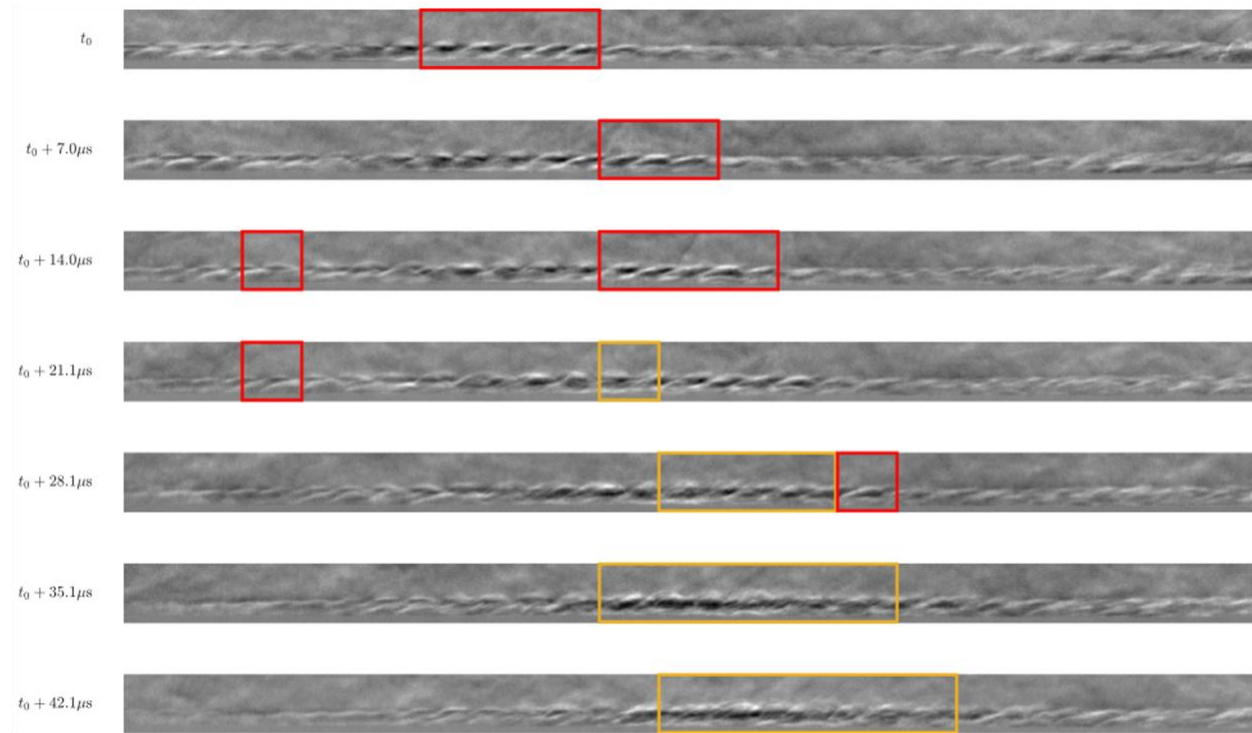
#### 4.2. 2<sup>nd</sup> Mode Instability to Turbulence Breakdown

(CUI) As noted in (Shumway *et al.* 2015), while turbulence often develops upstream and out of the field of view, our understanding of the breakdown mechanism from second-mode wave packets to turbulence is especially benefited by the analysis of image sequences where this breakdown occurs in frame. In the same work, Shumway uses MUSIC second-mode instability identification and high-frequency turbulence identification to illustrate automated identification of breakdown sequences. In this subsection, we demonstrate the potential for our combined turbulence and second-mode instability models to replicate these sequence identifications.

(CUI) Following the procedure outlined in Section 2.2, turbulence and second-mode instabilities were first classified in 3,000 sequential frames from the Langley Re9.7 experiment. These 3,000 frames were sampled from a portion of the video roughly in between where turbulence training data and second-mode instabilities training data were gathered. This lack of overlap between datasets resulted in the models appearing to under-classify features compared to their performance as discussed in Section 3, but they were still robust enough to service our applications. The importance of in-distribution sampling, which has the potential to address this accuracy concern, is touched on below and in the following subsection. These classifications were then passed into an algorithm which searched through the frames from last to first. Once the algorithm reached a turbulent slice in these frames, it tracked that section of the flow back in time and space based on an estimation of the propagation speed, first looking for earlier turbulent slices, then



wave packet slices, with an allowance for gaps in detection to increase robustness and compensate for transitional structures.

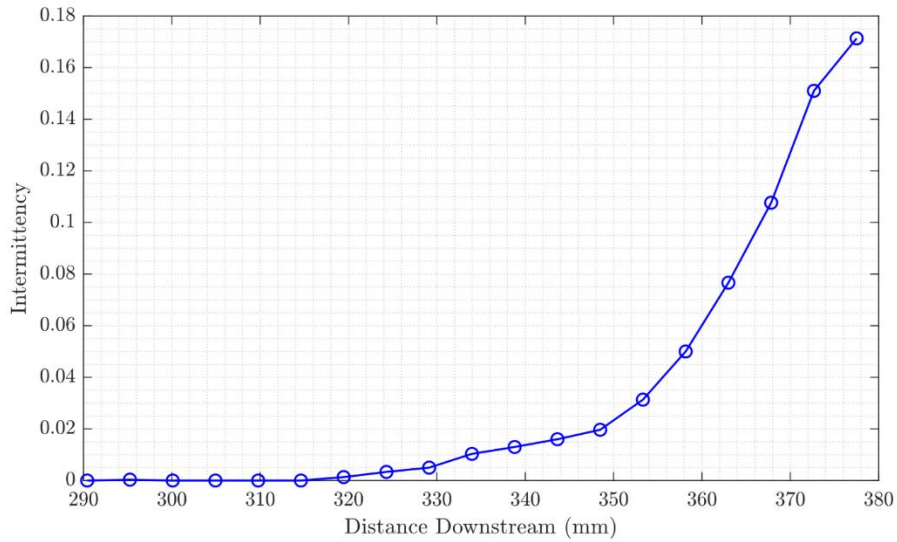


**(CUI) Figure 8: Identified sequence of a second-mode wave packet breaking down into turbulence. The red boxes represent slices classified as wave packets and the yellow boxes represent slices classified as turbulence.**

(CUI) Figure 8 demonstrates a successful example of automatically detecting and tracking a second-mode wave packet as it breaks down into turbulence. In  $t_0$ , the linear wave packet is well-identified. As it develops and begins to break down, however, the classifier fails to classify the nonlinear growth as a wave packet. When the packet has broken down and matured into more developed turbulence, the classifier successfully identifies and localizes the turbulence. In the continuous development from nonlinear growth to early turbulence, the classifiers fail to detect these intermediate structures. This is likely to our labeling and subsequent training, where we only labeled fully developed turbulence and linear growth wave packets. Such specificity to in-distribution structures is characteristic of transfer learning approaches (Weiss *et al.* 2016). Therefore, successful continuous tracking of intermediate breakdown forms may require more generalized labeling or additional classifiers.

(CUI) In addition, automated turbulence detection can help characterize the unsteady nature of the onset of turbulence by averaging detection results over time in an intermittency plot. Accordingly, the processed results from the 3,000 Langley Re9.7 frames were used to produce Figure 9 by calculating the percentage of classifications for which a slice was turbulent. The downstream location for a slice was determined by the center location of the slice. Data on the length of the cone visible in the schlieren images for Langley Re9.7 can be found in (Sousa *et al.* 2024).





**(CUI) Figure 9: Turbulence intermittency for Langley Re9.7.**

(CUI) These results are in agreement with similar experiments conducted in (Casper *et al.* 2014), which used pressure measurements and Canny edge detection on a  $7^\circ$  half-angle cone at  $Re = 9.75 \times 10^6 \text{ m}^{-1}$  to calculate turbulence intermittency. Under those similar test conditions, both measurements begin to see turbulence intermittency grow around 350-360 mm, showing consensus with our results where meaningful growth of turbulence intermittency begins around 350 mm.

### 4.3. Towards a Global Classification Model

(CUI) A robust classifier that can be deployed across diverse experimental data sets without requiring initial manual training would substantially reduce total overall classification time. Unlike analytical methods, however, deep classification is a black-box process and highly problem-specific (Weiss *et al.* 2016), making generalization or extending classification between experiments quite difficult. Therefore, experimenting with pre-processing techniques and their influence on classification can elucidate this black-box decision process. We postulate that classification is dependent on wavelength and boundary layer thickness relative to the slice dimensions. To test this, an image normalization algorithm was developed to standardize data across experimental configurations and conditions. As demonstrated in Section 3, the model performs well when deployed on the data it was originally trained on, i.e. in-distribution, but these experiments help determine what features the classifier may reason over. As we will demonstrate, our results indicate that aligning physical characteristics of testing data to the model's original training data significantly improves classification accuracy across experiments.

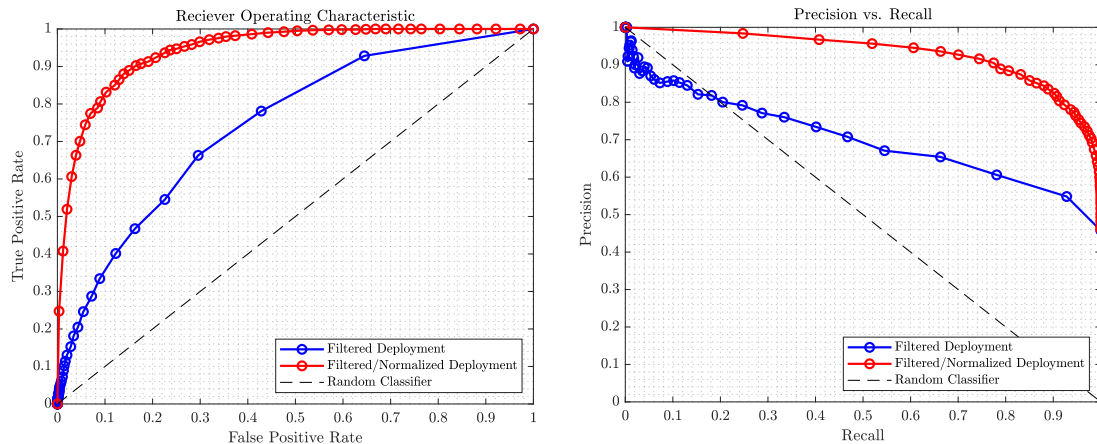
(CUI) The developed algorithm normalizes data sets based on three key parameters: boundary layer height, second-mode wave packet wavelength, and total image height. After normalization, these values are scaled to 9, 18, and 30 pixels, respectively. The boundary layer height of the unprocessed data is first found using the algorithm outlined in (Shumway *et al.* 2015). To determine the unprocessed data set's average wave packet wavelength, a Fourier Transform is applied to a single mean-subtracted image that contains a second-mode wave packet. The dominant frequency between twice and four times the boundary layer height is identified. The process is repeated for all images in the data set that contain a second-mode wave packet, and the mean of the dominant frequencies is calculated. The unprocessed wavelength is taken to be the reciprocal of this value. Prior to normalization, images are cropped such that no cone or excess space is

visible within the frame. To standardize the images to the desired parameters, each image is scaled by the ratio of the unprocessed boundary layer height to 9 in the surface-normal direction, and by the ratio of the unprocessed second-mode wave packet wavelength to 18 in the streamwise-direction. To maintain a composition with 30% boundary layer, the image is cropped or the top row of pixels in the image is duplicated accordingly, so the image reaches a height of 30 pixels. The images are also all flipped if needed to ensure that downstream is visualized to the right and upstream to the left.

(CUI) This normalization algorithm was applied to all three groups of experimental data. As a baseline for evaluating cross-experimental accuracy, the model architecture described in Section 2 was first trained on bandpass filtered Langley Re7.7 data and deployed on bandpass filtered Tunnel 9 and Cone Flare Re3.3 data. Another model was then trained on bandpass filtered and normalized Langley Re7.7 data and deployed on bandpass filtered and normalized Tunnel 9 and Cone Flare Re3.3 data. A comparison of the results between the two models and their tests is shown in Table 5.

(CUI) **Table 6: Bandpass Filtered Langley Re7.7 Model Performance and Bandpass Filtered and Normalized Langley Re7.7 Model Performance.**

Experiment, Re	Normalized	Accuracy	Precision	Recall
Tunnel 9, 3.6		54.35%	100%	0.2344%
Tunnel 9, 3.6	✓	72.41%	98.73%	10.71%
Cone Flare, 3.3		95.79%	Undefined	0%
Cone Flare, 3.3	✓	77.43%	5.352%	26.11%



(CUI) **Figure 10: ROC and PR curves without and with boundary-layer and wavelength normalization when the classification model is trained on filtered Langley Re7.7 images and tested on out-of-distribution Tunnel 9 images. Left plots the ROC curves, right plots the PR curves.**

(CUI) As seen in Table 5, normalization greatly improves a classifier when applied across experimental setups. For instance, when Tunnel 9 data was normalized and tested on the model trained on normalized Langley Re7.7 data, there was an 18% increase in overall classification accuracy compared to the unprocessed data. This improvement can be attributed to the fact that without normalization, the classifier exhibited a false positive rate of 0%, indicating the model failed to generalize and was overly conservative, labeling nearly all cases as negative. The classifier improvement can also be seen in the ROC and PR curves in Figure 10. In contrast, when the bandpass filtered and normalized Langley-trained model was deployed

on the bandpass filtered and normalized Cone Flare Re3.3 data, the accuracy decreased; however, the classifier itself showed meaningful improvement. While this seems counter-intuitive, the unprocessed Langley model classified nothing. Using the normalized model, on the other hand, produced some non-trivial identification, as realized by the increases in both precision and recall, marking an improvement in classification across experimental campaigns.

(CUI) In both cases, normalization of the second-mode characteristics to the slice dimensions improved classification efficiency, but the performance is still far from training within their own distributions (e.g. a model trained and deployed on its own experiment). Therefore, part of the classification performance is dependent on the structural features of the image, but a nontrivial part is based on other non-obvious image features. This may include signal-to-noise, features beyond the boundary layer, or subtle nonlinear differences in second-mode structural characteristics between differing flow conditions. While in-distribution classification shows significant success, this opaque classification process merits consideration when determining if this is the right classifier for a new application.

#### 4.4. Transfer Learning Classification Limitations

(CUI) Boasting accuracies beyond 90% with high precision and recall, in-distribution classification with a transfer learning approach has demonstrated success in identifying structural flow features in hypersonic flow. While this performance has exceeded state-of-the-art classification algorithms (Shumway *et al.* 2015), we note some practical limitations for future researchers to weigh when implementing this algorithm in their own analysis. Practically, this classification technique requires hand-labeling at least 100-150 frames for successful classification development, taking an individual researcher around one to two hours. This barrier may be too time-consuming and preclude application to simple analyses, but for whole video analyses (thousands of frames), it is still a reasonable option. Furthermore, state-of-the-art methods are not free of overhead work, and often require parametric tuning to elicit reasonable results. Beyond labeling, classification itself is a computationally expensive process. To assess with respect to the state-of-the-art, we measured time-to-classify 500 test frames and compared with state-of-the-art classification algorithms in Tables 7 and 8. While the transfer learning approach is quite high (about 1 sec a frame), its classification time is still comparable to many of the existing methods.

**(CUI) Table 7: Computational time for second-mode wave packet classification and localization of our transfer learning approach compared with existing methods. Values are normalized to the MUSIC computational time to provide a machine-agnostic sense of scale between the approaches.**

Experiment, Re	MUSIC	Auto-Corr	Filtered Image	STFT	Wavelet	Ours
Cone Flare, 3.3	1.00	6.87	0.75	0.68	1.04	0.85
Langley, 7.7	1.00	10.44	1.22	0.64	1.23	1.19
Tunnel 9, 3.6	1.00	7.21	1.13	0.72	0.83	1.05

(CUI) Researchers deploying this approach in their own work must be cognizant of some practical considerations as well. First, we note that the classifier works best when the video data is in-distribution, or the training data is a randomly sampled subset, and good representation, of the video. A known limitation of transfer learning is its sensitivity to classifying only within the distribution it's trained on (Weiss *et al.* 2016). Next, our experiments revealed that classification performance depends on the structure dimension relative to the slice, but this alone does not completely determine performance. This is inherent to many machine-learning processes, where the underlying decision-making process is intractably complex and

users must rely on stress-testing the model to observe its behavior (Weiss *et al.* 2016). While transfer learning appears to successfully work on our test sets, there are no tractable guarantees of performance we can make, and recommend users label additional validation images to stress-test their model. Even so, researchers must be careful when using transfer learning classification in conjunction with sensitive analyses that require perfectly classified structures. Overall, transfer learning demonstrates superior results compared to the state-of-the-art and the limitations are reasonable enough to apply this approach to future experimental campaigns.

**(CUI) Table 8: Computational time for developed turbulence classification and localization of our transfer learning approach compared with existing methods. Values are normalized to the STFT computational time to provide a machine-agnostic sense of scale between the approaches.**

Experiment, Re	STFT	Canny	Ours
Cone Flare, 4.5	1.00	0.93	0.63
Langley, 9.7	1.00	0.96	0.43

## 5. CONCLUSIONS AND FUTURE WORK

(CUI) In this study, we used transfer learning to classify and localize various hypersonic boundary layer features in schlieren imagery. We developed and deployed an architecture built with ResNet50 and a top layer dense neural network, which was trained and tested on six different experimental setups. Our workflow consisted of processing images via mean-subtraction and bandpass filtering, then splitting these images into smaller square segments to be fed into the model. Typical training sets of 100-200 images would fully train in just a few minutes, to then be deployed on larger sets of data to validate performance or accomplish more sophisticated analytical tasks. The model consistently demonstrated high accuracy, achieving over 90% global accuracy in identifying linear second-mode wave packets for all six conditions, and over 94% for half of those runs (see Table 2). We then applied the same workflow to classify fully developed turbulence in schlieren images, incorporating the exact same network architecture. The performance of this model was both highly accurate and consistent across three different data sets, achieving at least 97% accuracy in every case.

(CUI) We compared the performance of an ML approach to various other state-of-the-art identification techniques (Table 3, 4). For second-mode waves, our model always achieved higher accuracy and precision rates than all five other methods, and almost always showed higher recall. Similarly, for turbulence, our ML classifier demonstrated significantly higher accuracy, precision, and recall than the other existing methods for all experiments. Our results demonstrate that transfer learning is not only a viable option for boundary layer flow structure identification in schlieren images, but also consistently outperforms existing techniques.

(CUI) Finally, we extended the use of machine learning to other hypersonic flow analysis applications, including second-mode wave packet propagation speed, turbulence breakdown, and intermittency, which require the classification of much larger data sets (2,000+ images). Additionally, we explored the potential for a global classification model that could be applied to any hypersonic schlieren video to identify second-mode wave packets. Through a process of normalization, we were able to achieve improved results when training a model on a completely different experiment than it was tested on, indicating the potential for universal classification with a single model. Although the time to manually label the training data is not unreasonable, and often times proves quicker than parameter tuning for other existing methods, having a

*U.S. Journal for Hypersonics* (2026), Vol. 2, pp. 1 - ???.  
 ISSN: #####-##### , Electronic DTIC. Printed in the United States.

model that could instantly be applied to a new data set and perform reliably would allow for even quicker and larger-scale hypersonic analysis.

(CUI) We have identified a few avenues for future research. First, our experiments showed that classification does depend on the geometry of flow structures for detection, but not entirely so. Human identification, in contrast, looks almost solely for these features for classification. The transfer learning approach has clearly identified additional, subtle features in the image to perform classification that may be unphysical. In particularly sensitive experiments where precise results are crucial, understanding this decision-making process is critical, and we recommend further research in this area. Similarly, we noted that the classifiers work exceptionally well on linear second-mode wave packets and fully developed turbulence, due in large part to labeling only those features. The continuous, intermediate features were often under identified, and future research may develop a more continuous tracking classifier. This could allow researchers to better identify and study the nonlinear growth and early breakdown regimes. Finally, our work was conducted on flat, conical, axisymmetric hypersonic models, but we recommend researchers apply our model to more complex geometries, additional flow instabilities (e.g. first-mode wave packets), or more complicated flow conditions, like changing air properties. This extension could help improve our process and enable more sophisticated analyses for complicated flow conditions that better representative of hypersonic vehicles in practice.

## 6. ACKNOWLEDGEMENT

We would like to acknowledge Joseph Vasile of DEVCOM at the U.S. Army Research Laboratory for his insight into and helpful conversations on hypersonic boundary layers. Funding is provided by U.S. Army Research Office under the National Security Scholars Summer Internship Program.

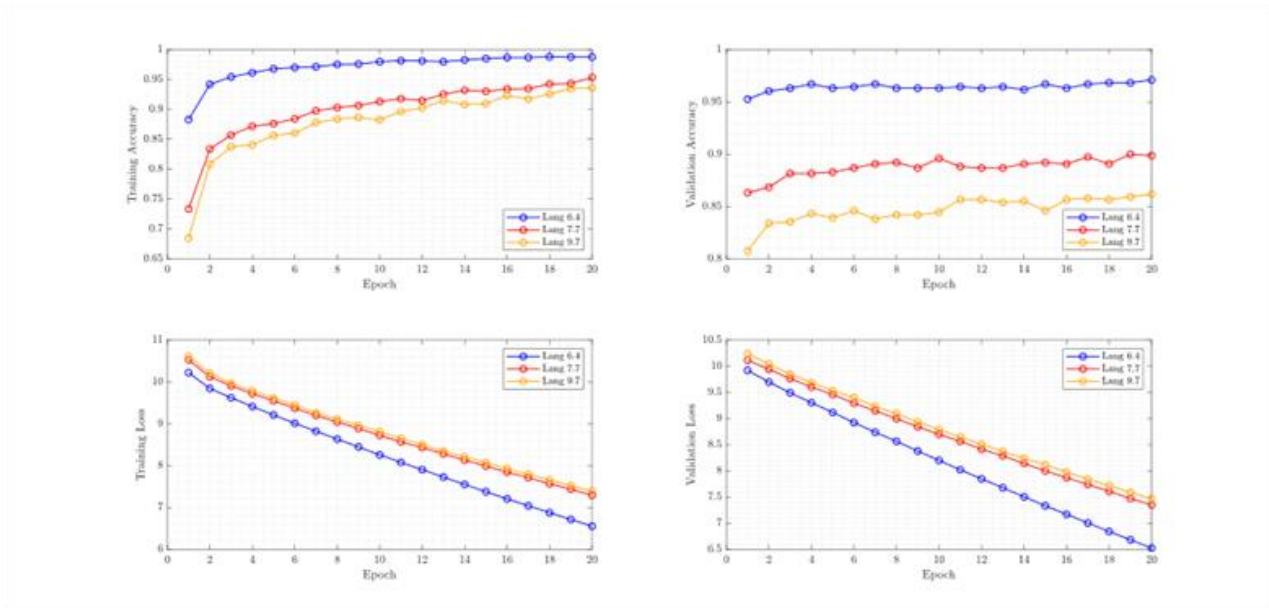
## 7. REFERENCES

- Alrowais, Fadwa, et al. "Deep transfer learning enabled intelligent object detection for crowd density analysis on video surveillance systems." *Applied Sciences*, vol. 12, no. 13, 30 June 2022, p. 6665, <https://doi.org/10.3390/app12136665>.
- Alzubaidi, Laith, et al. "Novel transfer learning approach for medical imaging with limited labeled data." *Cancers*, vol. 13, no. 7, 30 Mar. 2021, p. 1590, <https://doi.org/10.3390/cancers13071590>.
- Butler, Cameron S., and Stuart J. Laurence. "Transitional hypersonic flow over slender cone/flare geometries." *Journal of Fluid Mechanics*, vol. 949, 30 Sept. 2022, <https://doi.org/10.1017/jfm.2022.769>.
- Casper, Katya M., et al. "Transition statistics measured on a 7-degree hypersonic cone for turbulent Spot Modeling." *52nd Aerospace Sciences Meeting*, 10 Jan. 2014, <https://doi.org/10.2514/6.2014-0427>.
- Cheney, Margaret. "The linear sampling method and the music algorithm." *Inverse Problems*, vol. 17, no. 4, 23 July 2001, pp. 591–595, <https://doi.org/10.1088/0266-5611/17/4/301>.
- Chollet, Francois. "Keras-Team/Keras: Deep Learning for Humans." *GitHub*, 2015, [github.com/keras-team/keras](https://github.com/keras-team/keras).
- Davis, Jesse, and Mark Goadrich. "The relationship between precision-recall and ROC curves." *Proceedings of the 23rd International Conference on Machine Learning - ICML '06*, 2006, pp. 233–240, <https://doi.org/10.1145/1143844.1143874>.
- Davis, Jesse, and Mark Goadrich. "The relationship between precision-recall and ROC curves." *Proceedings of the 23rd International Conference on Machine Learning - ICML '06*, 2006, pp. 233–240, <https://doi.org/10.1145/1143844.1143874>.
- Demetriades, Anthony. "Hypersonic viscous flow over a slender cone. III - laminar instability and transition." *7th Fluid and Plasma Dynamics Conference*, 17 June 1974, <https://doi.org/10.2514/6.1974-535>.

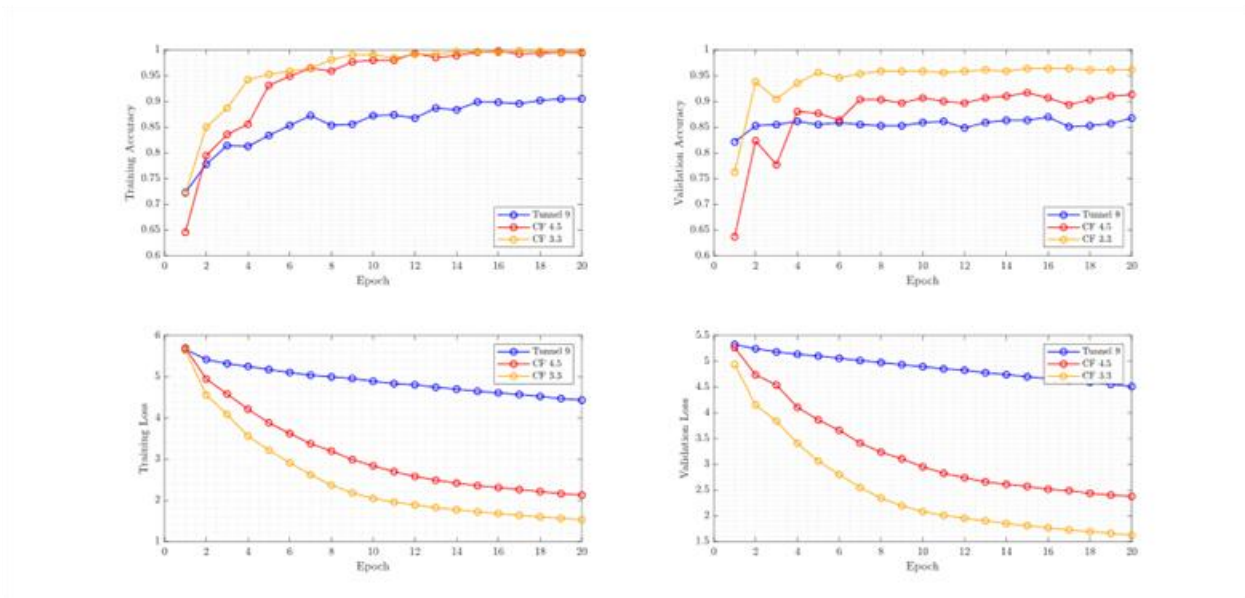
*U.S. Journal for Hypersonics* (2026), Vol. 2, pp. 1 - ???.  
 ISSN: ##### , Electronic DTIC. Printed in the United States.

- Farge, M. "Wavelet transforms and their applications to turbulence." *Annual Review of Fluid Mechanics*, vol. 24, no. 1, 1 Jan. 1992, pp. 395–457, <https://doi.org/10.1146/annurev.fluid.24.1.395>.
- Fedorov, Alexander. "Transition and stability of high-speed boundary layers." *Annual Review of Fluid Mechanics*, vol. 43, no. 1, 21 Jan. 2011, pp. 79–95, <https://doi.org/10.1146/annurev-fluid-122109-160750>.
- He, Kaiming, et al. "Deep residual learning for image recognition." *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 770–778, <https://doi.org/10.1109/cvpr.2016.90>.
- Kennedy, Richard E., et al. "Investigation of the second-mode instability at mach 14 using calibrated Schlieren." *Journal of Fluid Mechanics*, vol. 845, 20 Apr. 2018, <https://doi.org/10.1017/jfm.2018.269>.
- Kennedy, Richard E., et al. "Visualization of the second-mode instability on a sharp cone at mach 14." *2018 AIAA Aerospace Sciences Meeting*, 7 Jan. 2018, <https://doi.org/10.2514/6.2018-2083>.
- Kim, Hee E., et al. "Transfer learning for medical image classification: A literature review." *BMC Medical Imaging*, vol. 22, no. 1, 13 Apr. 2022, <https://doi.org/10.1186/s12880-022-00793-7>.
- Krizhevsky, Alex, et al. "ImageNet classification with deep convolutional Neural Networks." *Communications of the ACM*, vol. 60, no. 6, 24 May 2017, pp. 84–90, <https://doi.org/10.1145/3065386>.
- Laurence, Stuart J., et al. "Schlieren-based techniques for investigating instability development and transition in a hypersonic boundary layer." *Experiments in Fluids*, vol. 55, no. 8, 23 July 2014, <https://doi.org/10.1007/s00348-014-1782-9>.
- Laurence, Stuart J., Alexander Wagner, and Klaus Hannemann. "Experimental study of second-mode instability growth and breakdown in a hypersonic boundary layer using high-speed schlieren visualization." *Journal of Fluid Mechanics* 797 (2016): 471–503.
- LeCun, Yann, and Yoshua Bengio. "Convolutional networks for images, speech, and time series." *The Handbook of Brain Therapy and Neural Networks*, 1 Oct. 1998, pp. 225–258.
- LeCun, Yann, Fu Jie Huang, et al. "Learning methods for generic object recognition with invariance to pose and lighting." *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, vol. 2, 19 July 2004, pp. 97–104, <https://doi.org/10.1109/cvpr.2004.1315150>.
- LeCun, Yann, Yoshua Bengio, et al. "Deep learning." *Nature*, vol. 521, no. 7553, 27 May 2015, pp. 436–444, <https://doi.org/10.1038/nature14539>.
- Lin, Jianzhe, et al. "Active-learning-incorporated Deep Transfer Learning for hyperspectral image classification." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 11, Nov. 2018, pp. 4048–4062, <https://doi.org/10.1109/jstars.2018.2874225>.
- Mascarenhas, Sheldon, and Mukul Agarwal. "A comparison between VGG16, VGG19 and Resnet50 architecture frameworks for Image Classification." *2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON)*, 19 Nov. 2021, pp. 96–99, <https://doi.org/10.1109/centcon52345.2021.9687944>.
- Oquab, Maxime, et al. "Learning and transferring mid-level image representations using convolutional neural networks." *2014 IEEE Conference on Computer Vision and Pattern Recognition*, June 2014, pp. 1717–1724, <https://doi.org/10.1109/cvpr.2014.222>.
- Shumway, Nathan, and Stuart J. Laurence. "Methods for identifying key features in Schlieren images from hypersonic boundary-layer instability experiments." *53rd AIAA Aerospace Sciences Meeting*, 3 Jan. 2015, <https://doi.org/10.2514/6.2015-1787>.
- Sousa, Cole E., et al. "Global Analysis of nonlinear second-mode development in a Mach-6 boundary layer from high-speed Schlieren Data." *Experiments in Fluids*, vol. 65, no. 2, 31 Jan. 2024, <https://doi.org/10.1007/s00348-023-03758-w>.
- Wang, Weimin, et al. "Breast cancer image classification method based on Deep Transfer Learning." *Proceedings of the International Conference on Image Processing, Machine Learning and Pattern Recognition*, 13 Sept. 2024, pp. 190–197, <https://doi.org/10.1145/3700906.3700937>.
- Weiss, Karl, et al. "A survey of Transfer Learning." *Journal of Big Data*, vol. 3, no. 1, 28 May 2016, <https://doi.org/10.1186/s40537-016-0043-6>.

## APPENDIX A: MODEL TRAINING ACCURACY AND LOSS PLOTS

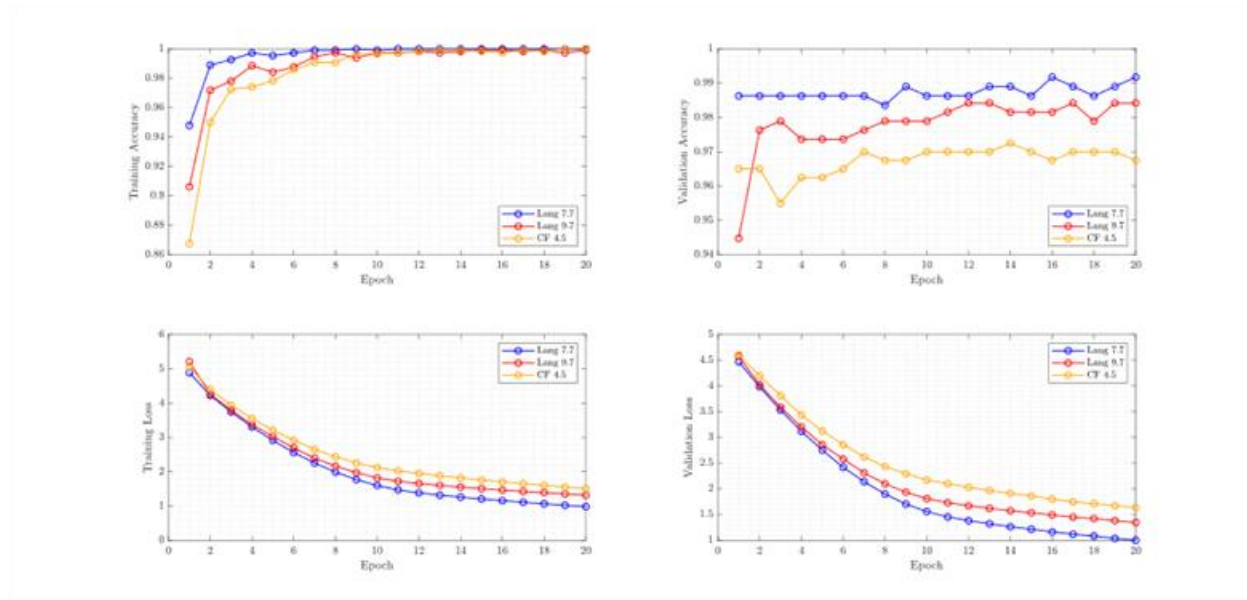


(CUI) Figure A1: Langley Re6.4, Langley Re7.7, and Langley Re9.5 second-mode wave packet classification model training curves.



(CUI) Figure A2: Tunnel 9, Cone Flare Re4.5, and Cone Flare Re3.3 second-mode wave packet classification model training curves.





**(CUI) Figure A3: Langley Re7.7, Langley Re9.5, and Cone Flare Re4.5 turbulence classification model training curves.**