

Edge Detection and Machine Learning Approach to Identify Flow Structures on Schlieren and Shadowgraph Images *

Irina Znamenskaya¹[0000-0001-6362-9496], Igor Doroshchenko²[0000-0002-0488-0020], and Daria Tatarenkova³[0000-0002-2591-850X]

Faculty of Physics, Lomonosov Moscow State University, 119991, Moscow, Russia

¹ znamen@phys.msu.ru

² doroshenko.igor@physics.msu.ru

³ di.tatarenkova@physics.msu.ru

Abstract. Schlieren, shadowgraph and other types of refraction-based techniques have been often used to study gas flow structures. They can capture strong density gradients, such as shock waves. Shock wave detection is a very important task in analyzing unsteady gas flows. High-speed imaging systems, including high-speed cameras, are widely used to record large arrays of shadowgraph images. To process large datasets of the high-speed shadowgraph images and automatically detect shock waves, convective plumes and other gas flow structures, two computer software systems based on the edge detection and machine learning with convolutional neural networks (CNN) were developed. The edge-detection software utilizes image filtering, noise removing, background image subtraction in the frequency domain and edge detection based on the Canny algorithm. The machine learning software is based on CNN. We developed two neural networks working together. The first one classifies the image dataset and finds images with shock waves. The other CNN solves the regression task and defines shock wave position (single number) based on image pixels tensor (3-D array of numbers) for each image. The supervised learning code based on example input-output pairs was developed to train models. It was shown, that the machine learning approach gives better results in shock wave detection accuracy, especially for low-quality images with a strong noise level. Software system for automated shadowgraph images processing and x-t curves of the shock wave and convective plume movement plotting was developed.

Keywords: Flow Visualization, Image Processing, Shock Wave Detection, Machine Learning, CNN, Transfer Learning, Image Edge Detection, Classification, Regression, Convective Plume.

Copyright © 2020 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

* This work has been supported by the Russian Science Foundation (Grant number 18-19-00672).

1 Introduction

A shock wave is a discontinuity surface that moves faster than the local speed of sound in the medium. The pressure, temperature and density change abruptly on that surface. Shock waves and other flow structures detection on the experimental and CFD data sets are very important tasks. High-speed digital cameras and CFD software generate large amounts of data, thus special software systems for detecting shock waves and other different flow structures are required. Modern CFD shock wave detection methods include gradient maxima-based methods, normal Mach number-based methods and also characteristics methods [1, 2]. The characteristics method is very precise, but its implementation is much more difficult than the other ones. The numerical oscillation and dissipation make it harder to detect shocks [1].

Schlieren and shadowgraph techniques have been used since the 19th century to visualize density fields in the transparent media. Both methods are based on the refraction phenomenon [3]. The high-speed schlieren and shadowgraph techniques are widely used to investigate unsteady gas flows with shock waves and other flow structures. Modern digital high-speed cameras can record videos with the frame rate up to 10 000 000 frames / s, thus it is important to develop software systems for automated flow structures recognition and measurement. Today, the most promising approaches to solve these problems are based on digital image processing with different edge detection and object recognition algorithms [4, 5] and using machine learning [6, 7, 8] to identify complex flow structures. Different image edge detection algorithms are suitable for shock wave detection [9], including Prewitt, Roberts, Sobel and Canny [10]. It was shown by different researchers that the Canny algorithm best suited for schlieren and shadowgraph image processing [9, 5]. Cui et al. [9] built a shock wave detection software to measure shock stand-off distance from the model inside a supersonic wind tunnel. The software utilizes different edge detection algorithms, but Laplacian of Gaussian (LoG) and Canny methods showed the most precise results, especially if the image sequence has a high noise level. Li et al. [5] made software for shock wave detection and tracking. A modified version of the Canny algorithm was developed. To improve image quality, the software includes background image subtraction in the frequency domain and additional filtering before edge detection. Estruch et al. [11] developed shock-detection software based on a modified Canny edge detection algorithm. Shock wave/turbulent boundary-layer interaction was recorded using a high-speed schlieren optical system. The dynamics of the shock wave was measured by means of the developed software. It was shown, that background image subtraction in the frequency domain using Fast Fourier Transform (FFT) removes noise more efficiently than that in the space domain. Fujimoto et al. [12] successfully applied the Canny edge detection method to CFD solutions by replacing pixel brightness with the pressure values inside the CFD grid.

The machine learning-based approach for identifying flow structures on schlieren images is also rapidly developing. It was shown, that a deep learning approach is suitable for the identification of shock waves in large tensor field datasets and it can effectively capture shock features [13]. Dehghan Manshadi et al. [14] proposed an image classification and processing system for schlieren images of an object in the wind

tunnel. The system was able to extract three features from the images: refraction angle of the bow shock, its line intensity difference and mean width of the line. Based on these features, the classification system calculates the flow speed near the model. Neural networks can extract not only shock waves, but also any other flow patterns. For example, Colvert et al. [15] successfully applied a neural network to classify vortex wakes behind an airfoil.

In the present study, we developed two software systems for the automatic processing of large shadowgraph image data sets. The first one is based on MATLAB[®] and built-in Image Processing Toolbox. We developed a modified Canny edge detection algorithm in order to identify shock waves moving in a shock tube channel and convective plumes spreading from the surface pulsed gas discharges. The second program is a web application written in JavaScript and using the open-source TensorFlow.js and ml5.js machine learning libraries. We used CNN trained to detect shock waves and to give its locations. The software automatically processes images in a given folder and plots the x-t curves of the shock wave movement and allows users to export the results in the .csv format.

2 Experiments

2.1 Experimental setup

The experimental setup [16, 17] is a shock tube with the discharge chamber built in its low-pressure section. The tube has a rectangular cross-section of $24 \times 48 \text{ mm}^2$. The discharge section has the same cross-section and a length of 100 mm. It has a complex discharge arrangement for pulse surface and volume discharge creation. The sidewalls of the discharge chamber are made of quartz glass for optical access. We performed high-speed shadowgraph imaging of the flat shock wave moving in a channel without the discharge in the first set of experiments (see Fig. 1) and the blast waves spreading from the surface discharge channel in the second set of experiments (see Fig. 2). The optical system used for high-speed shadowgraph imaging is described in detail in [18]. The recording frame rate is up to 525 000 frames / s. The camera imaging exposure is $1 \mu\text{s}$.

2.2 Flat shock wave moving in a channel

Shock wave forms after the breakdown of the diaphragm, separating high-pressure and low-pressure sections of the shock tube, due to the pressure difference. We use air as a working gas. The shock Mach number was $M = 2 - 3$ depending on the pressure in front of the shock wave. Fig. 1 illustrates the schematic of a flat shock wave moving in the rectangular tube channel. The discharge system is not involved in this set of experiments.

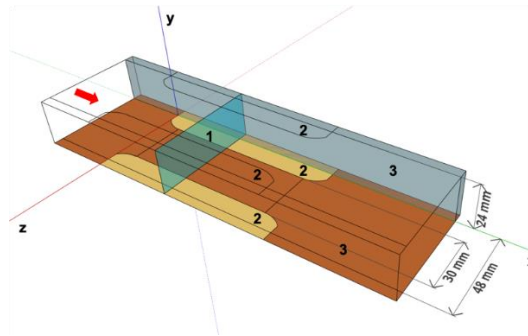


Fig. 1. Test section with a shock wave schematic. 1 – flat shock wave; 2 – electrodes; 3 – quartz glass windows. The red arrow indicates shock wave and the following gas flow direction. The top wall and the right window are drawn transparent for clarity

2.3 Shock wave structure and the convective plume created by the pulsed surface discharge

Blast waves and the following convective plumes are generated by the nanosecond sliding surface discharge on the top and on the bottom walls of the discharge chamber. We study the process on the bottom wall of the discharge section. The full electrical energy of the discharge is $E = 0.71$ J. The short energy deposition time and rapid gas heating ($t < 1 \mu\text{s}$) lead to the flow with shock waves formation. The discharge operates in quiescent air at low air pressure up to 100 Torr. The surface discharge covers an area of $100 \times 30 \text{ mm}^2$. The discharge energy is distributed uniformly across the discharges area with the rare local cylindrical high-energy channels. The uniformly distributed plasma sheet forms a flat horizontal shock wave moving up from the sliding discharge. Local high-energy cylindrical channels form semi-cylindrical shock waves and the following convective plumes. It was previously shown that the plume dynamics is explained by the forced convection, caused by the discharge-induced shock wave [19]. Fig. 2 illustrates the discharge arrangement and the generated shock waves.

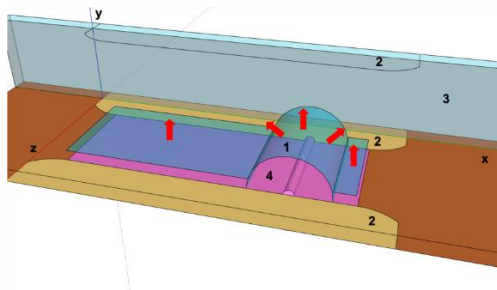


Fig. 2. Test section with a surface discharge (plasma sheet) and the generated shock waves. 1 – generated shock waves; 2 – electrodes; 3 – quartz glass windows; 4 – convective plumes; The red arrow indicates shock wave and the following gas flow direction. The top wall and the right window are drawn transparent for clarity

3 Shock wave detection based on Canny Edge Detection algorithm

3.1 Canny image edge detection algorithm review

Canny edge detection is one of the most advanced and popular edge detection algorithms. It was introduced by John F. Canny in 1986 [10]. The algorithm includes four image processing steps. The first one is the noise reduction with a low-pass Gaussian filter (see Eq. 1):

$$G(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2+y^2}{2\sigma^2}}. \quad (1)$$

where x is the distance from the given pixel in the horizontal axis, y – in the vertical, σ is the standard deviation of the Gaussian distribution. We use MATLAB[®] implementation of the Canny edge detection algorithm, which utilizes 16×16 Gaussian filter by default.

The next step is to find an image intensity gradient. There are a few approaches to do this. The simplest one is to directly calculate the pixel intensity derivatives in the x and y directions and to build a gradient vector:

$$\nabla I = \left[\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right]. \quad (2)$$

$$\frac{\partial I}{\partial x} = \lim_{h \rightarrow 0} \frac{I(x+h,y) - I(x,y)}{h} \approx I(x+1, y) - I(x, y), \quad (3)$$

$$\frac{\partial I}{\partial y} = \lim_{h \rightarrow 0} \frac{I(x,y+h) - I(x,y)}{h} \approx I(x, y+1) - I(x, y), \quad (4)$$

where x and y are pixels column and row numbers. The more advanced approaches utilize custom image gradient operators to compute an approximation of the gradient of the image intensity. For example, the Sobel operator [20] uses two 3×3 matrices which are convolved with the image intensity matrix:

$$I_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * I, \quad I_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * I. \quad (5)$$

Thus, the partial derivatives and the gradient's direction can be found as (6).

$$\frac{\partial I}{\partial x} \approx -I(x-1, y-1) - 2I(x-1, y) - I(x-1, y+1) + I(x+1, y-1) + 2I(x+1, y) + I(x+1, y+1), \quad (6)$$

$$\frac{\partial I}{\partial y} \approx -I(x-1, y-1) - 2I(x, y-1) - I(x+1, y-1) + I(x-1, y+1) + 2I(x, y+1) + I(x+1, y+1), \quad (7)$$

$$\theta = \text{atan}\left(\frac{I_y}{I_x}\right). \quad (8)$$

The edge direction is always normal to the gradient direction.

Canny edge detection method uses another approach by finding a derivative of the Gaussian filter (see Eq. 9):

$$I_x = I * \frac{\partial G}{\partial x}, \quad I_y = I * \frac{\partial G}{\partial y} \quad (9)$$

The third step is non-maximum suppression. The pixels which are not lying on the edges are removed. The fourth step is the hysteresis thresholding. On this step, we set two threshold values: minimum threshold and a maximum threshold. The edges with intensity gradient higher than the maximum threshold are sure to be edges and the edges below the minimum threshold value are treated as non-edges. If the edge intensity gradient is between these two threshold values, such edges are classified based on their connectivity. If such an ‘‘intermediate’’ edge is connected to the edge with intensity gradient higher than the maximum threshold value, the algorithm treats such edge as real. Otherwise, the edge gets discarded (its intensity value sets to 0). At the end, we get a binary black and white image with the black background and white edges (with possible pixel intensity values equal to 0 or 1).

In our code, we used MATLAB[®] implementation of the Canny algorithm. We selected various threshold and σ values manually depending on our tasks and shadow-graph images under study. We manually specified the maximum threshold value and the lower value was calculated automatically as the maximum threshold multiplied by 0.4.

3.2 Shock wave edge-detection

Fig. 3 shows the shadowgraph image of a shock wave moving in a channel and the corresponding light intensity distribution. The gas pressure ahead of the shock wave is 15 Torr. The shock Mach number is $M = 2.3$. The imaging frame rate is 525 000 frames/s.

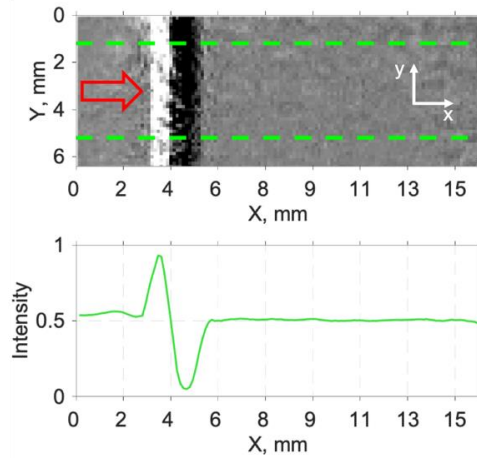


Fig. 3. Shadowgraph image of a shock wave and its light intensity profile. Green dashed lines indicate the intensity averaging band. The red arrow indicates shock wave and the following gas flow velocity direction

Thus, the shadowgraph image of a shock wave contains a light strip following the dark one. Such intensity pattern is a perfect target for edge detection algorithms due to significant intensity gradients. Fig. 4 illustrates shock wave shadowgraph image processing using Canny edge detection. Varying threshold and σ we selected the appropriate settings for different image sets. After processing all of the given images our software is able to automatically track shock positions and record their motion dynamics. Before applying edge detection, the software system can perform a background image subtraction in the frequency domain if the background image is provided.

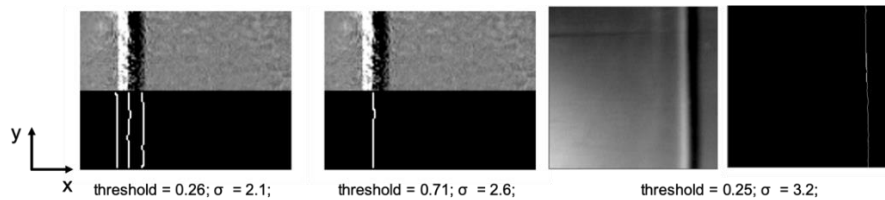


Fig. 4. Shadowgraph images with shock waves and the corresponding Canny edge detection processing. Shock wave moves from left to right

3.3 Convective plume edge-detection

The edge detection approach is also suitable for more complex tasks solutions. Fig. 5 illustrates the flow structure evolution created by the pulsed surface discharge configuration (see Fig. 2). Shock waves, compression waves, convective plumes with vortex structures were detected. Flat shock wave spreads from the plasma sheet surface and the cylindrical one is created by the local high-energy surface discharge channel. The physics of the convective plume is based on the forced convection, caused by the shock wave [19]. The air pressure of quiescent air was $p = 93$ Torr. The height of each image is 24 mm. The threshold values were from 0.1 to 2.2, $\sigma = 3.2$.

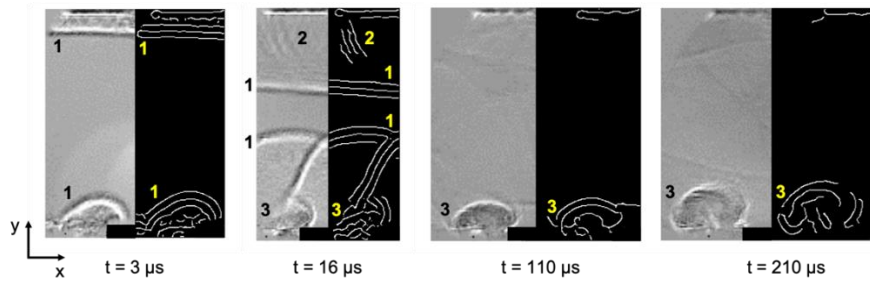


Fig. 5. A sequence of shadowgraph images of the gas flow, spreading from the pulsed surface discharges, and the corresponding Canny edge detection binary images. 1 – shock waves; 2 – compression waves; 3 – plumes with vortex structures

The MATLAB[®]-based software system for automatic plume size detection was developed. It allows us to process a large amount of data very quickly and to compare the results with theory or CFD. The algorithm applies edge detection to every frame in the selected folder and finds the top-most point of the plume. The y -position of this point is treated as plume's size at a given time. The software records measured data, plots the distance (plume's size) vs time dependency and makes an animation of the results.

A small part of one of the obtained sequences of shadowgraph images (200–400 images are in a full sequence) and their processing are illustrated in Fig. 6. The recording frame rate was 150 000 frames / s. Threshold value was set to 0.12; $\sigma = 1.3$. Air pressure was $p = 93$ Torr.

Fig. 7 shows the plume size vs time dependency and the polynomial approximation with the powers of 2 and 7. Each frame sequence has about 5%-8% wrong plume size detections due to the shock wave presence. Such points were automatically eliminated from the result by means of comparison of the difference between each point and the next one with the maximum physically possible step value. This value was set manually in the range between 5 and 10 pixels depending on the experimental conditions. The plume moves unevenly with variable speed and is well approximated by a seventh order polynomial. The second-order polynomial gives the mean plume size values.

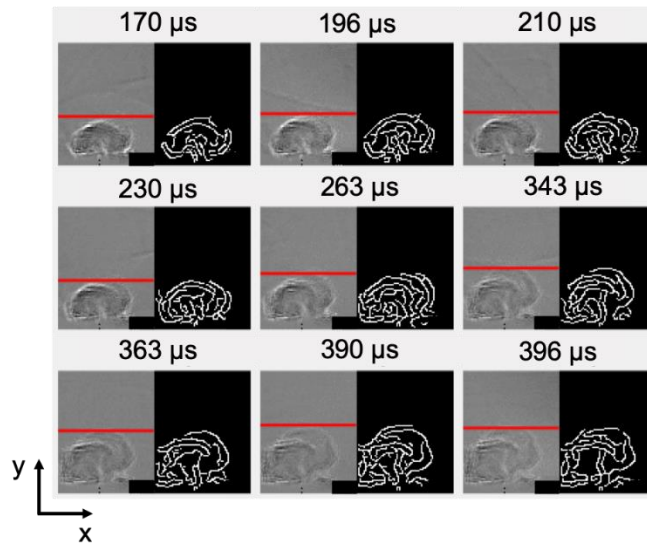


Fig. 6. A sequence of shadowgraph images of the convective plume evolution. The red horizontal line indicates the automatically-detected plume top-position

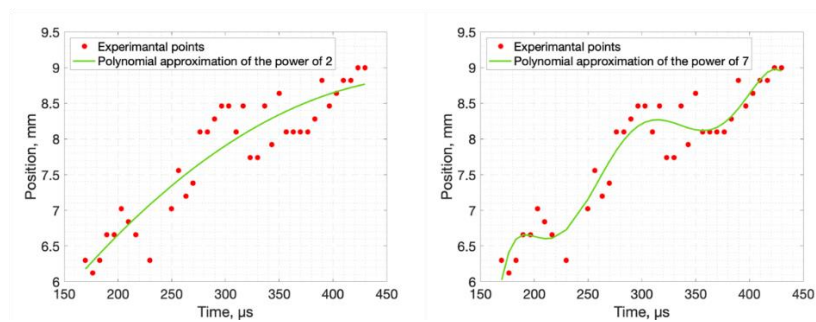


Fig. 7. Automatically detected plume size vs time plot and its polynomial approximations

4 Neural networks

4.1 Proposed software system

Edge detection algorithms are very useful in schlieren and shadowgraph image processing, but they have their flaws. Each image sequence requires different threshold

settings to get the desired result. If the image contains different objects with the same intensity gradients, for example, shock waves and plume borders, it is not possible to detect only one of the object types in the image. We tried to solve these problems by using neural networks and to create a more powerful flow structures detection system.

Our solution is based on the open-source machine learning libraries such as TensorFlow.js and ml5.js and it is written in JavaScript and PHP programming languages. It has a user-friendly GUI system with two main screens. The application has a learning screen and a processing screen. The learning screen is designed to create classification and regression models and save them on the user's computer. The processing screen is designed for shadowgraph image classification and flow structures detection using the pre-trained models. The schematic of the test process of the shock wave detection is illustrated in Fig. 8.

Thus, the developed software allows us to automatically process hundreds of shadowgraph images within a minute. Previously, it took several hours to process one sequence of such frames manually [19].

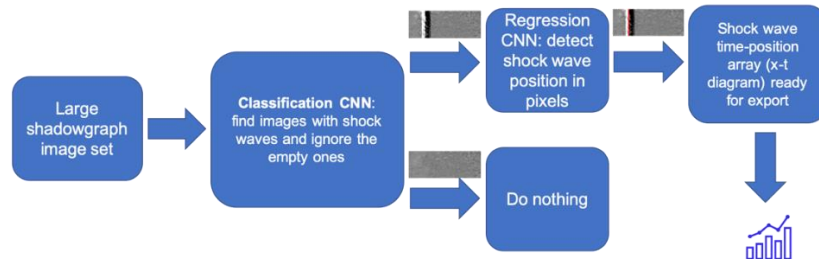


Fig. 8. Schematic of the shock wave detection process using neural networks

To detect the shock wave position, two CNNs are working together. The first one finds good-quality images with shock waves using a pre-trained classification model. The second one gets the images containing shocks as an input and returns the position of the shock based on the pre-trained regression model. It was shown, that the neural network approach gives better results, then the edge detection algorithm, when the image has poor quality and high noise level with intensity gradients comparable with that on the shock wave image.

4.2 Classification

We trained classification CNN to recognize 3 different image classes: empty image, an image with a shock wave and an image containing plume. We used up to 150 images of each class for training. The training process includes 50 epochs, the batch size was 16 and the learning rate was set to 0.001. Fig. 9 shows the accuracy and the loss plots. The accuracy is the percentage of classifications that a model gets during training. Loss

describes how well a model has learned to predict the right classifications for a given set of samples. They show a very good learning rate and the classification accuracy reaches 100% for the given samples.

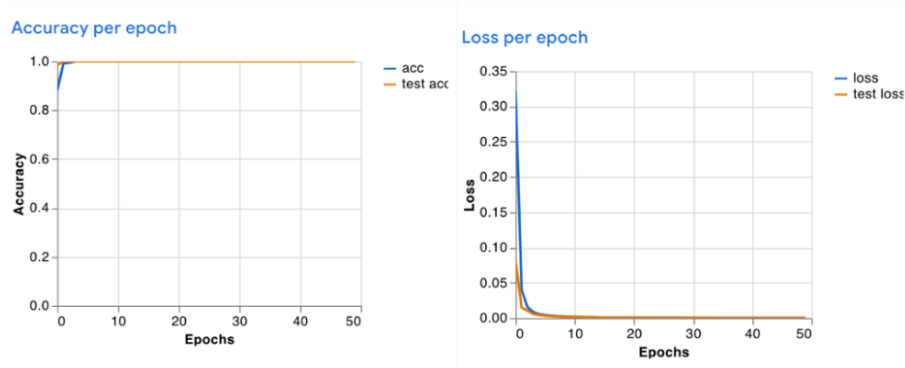


Fig. 9. Accuracy and loss vs epoch of the classification model

4.3 Regression

We used another CNN for shock wave position detection. The neural network was trained on input-output pairs of data. The images were passed to the input as 3-D tensors (the tensor's width and height equal to image width and height and its depth contains image color information). Output was a single number equal to the shock wave position in pixels. We used the pre-trained MobileNet [21] model to extract image features. MobileNet is a CNN model for image classification which does not require a lot of computational power to operate or to apply transfer learning. We used the Transfer Learning (TL) process to retrain the model with our shadowgraph images data. We trained the model to solve the regression task and to predict the shock wave position on the new, previously unseen images. Fig. 10 illustrates the learning efficiency for the 42 training samples (shadowgraph images containing shock wave). The learning rate is quite good for the given number of samples.

Fig. 11 shows the automatically detected shock wave locations and the corresponding distance vs time plot. The dotted line represents the linear trend line, which shows a good measurement accuracy.

The described regression-based learning approach has its flaws. It is based on the pre-trained model and the feature extraction for some of the complex flow structures may fail and the prediction gives up to 50% error results. It happens, for example, with the convective plume images on the later stages of plume evolution, when it splits into a few distinct plumes due to convection. Plume top-position detection for such images may give the undesired results. Thus, to solve these kinds of problems, we are going to build our model based on the shadowgraph image data from scratch in the future.

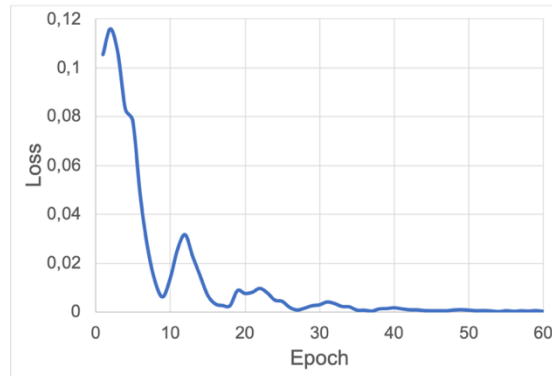


Fig. 10. Loss vs epoch of the regression model

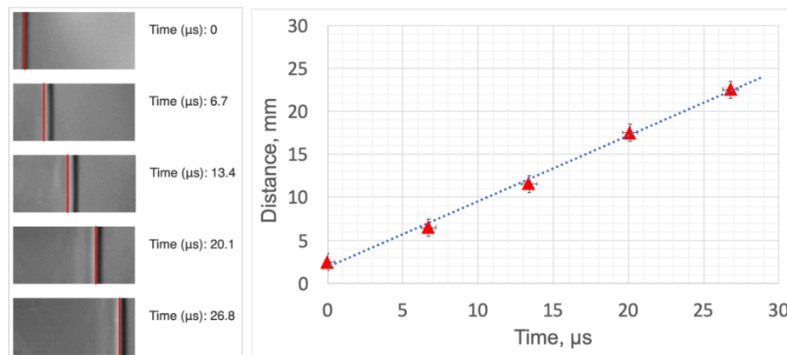


Fig. 11. Automatic shock wave detection based on the CNN

5 Conclusion and future works

Two image processing software systems for fluid structures detection were built, tested and used for real scientific problem solutions. The software can process large amount of schlieren and shadowgraph images obtained by the high-speed cameras in the experiments. First system is based on the Canny edge detection algorithm and allows to detect and track different gas flow structures visualized by the schlieren or shadowgraph techniques. Additionally, the edge-detection code implements noise removing, background image subtraction in frequency domain and image filtering before the edge detection. We successfully tested the software on images containing shock

waves and convective plumes. Shock wave dynamics and plume's size versus time were measured automatically for different experimental conditions.

The second software system was designed to solve the same tasks as the previous one using CNN. It includes user-friendly GUI and uses two neural networks to classify images, detect different types of gas flow structures and to track their movement. The classification CNN was built from scratch on the large shadowgraph image data set and gives perfect results in image classification with the accuracy of 100% for the given samples. It can distinguish 3 different flow structures at the moment: shock waves, background empty images with quiescent air and the convective streams. The second CNN is designed for flow structures tracking by solving the regression task. It works well for simple flow structures, such as shock waves. It was shown, that the CNN approach gives better results in shock wave detection than the edge detection when the background noise level is high. But our implementation of the flow-tracking CNN often mistakes while processing the complex flow structures.

In the future work we will train our classification model to be able to detect more gas flow structures, such as vortices, laminar-turbulent transition areas, supersonic jets, bow-shocks etc. The second CNN, used for shock tracking, will be rebuilt from scratch and trained on large amount of shadowgraph image data set instead of using pre-trained model and feature extraction approach. We will continue to improve our edge-detection software and make the user-friendly user interface.

6 Acknowledgements

This work has been supported by the Russian Science Foundation (Grant number 18-19-00672).

References

1. W. Ziniu, X. Yizhe, W. Wenbin and H. Ruifeng, "Review of shock wave detection method in CFD post-processing," *Chinese Journal of Aeronautics*, vol. 26, no. 3, pp. 501-513, 2013.
2. D. Lovely and R. Haimes, "Shock detection from computational fluid dynamics results," *AIAA-99-3285*, 1999.
3. G. S. Settles and M. J. Hargather, "A review of recent developments in schlieren and shadowgraph techniques," *Measurement Science and Technology*, vol. 28, no. 4, 2017.
4. J. Wolfram and J. Martinez Schramm, "Pattern Recognition in High Speed Schlieren Visualization at the High Enthalpy Shock Tunnel Göttingen (HEG)," in *New Results in Numerical and Experimental Fluid Mechanics VII. Notes on Numerical Fluid Mechanics and Multidisciplinary Design*, Berlin, 2010.
5. G. Li, M. Burak Agir, K. Kontis, T. Ukai and S. Rengarajan, "Image Processing Techniques for Shock Wave Detection and Tracking in High Speed Schlieren and Shadowgraph Systems," *Journal of Physics: Conference Series*, vol. 1215, 2019.
6. D. R. Edla, P. Lingras and Venkatanareshbabu K., *Advances in Machine Learning and Data Science: Recent Achievements and Research Directives*, Springer, 2018.
7. Y. Liu, Y. Lu, Y. Wang, D. Sun, L. Deng, F. Wang and Y. Lei, "A CNN-based shock detection method in flow visualization," *Computers & Fluids*, vol. 184, pp. 1-9, 2019.

14 I. Znamenskaya, I. Doroshchenko, D. Tatarenkova

8. S. Ye, Z. Zhang, X. Song, Y. Wang, Y. Chen and C. Huang, "A flow feature detection method for modeling pressure distribution around a cylinder in non-uniform flows by using a convolutional neural network," *Scientific Reports*, vol. 10, 2020.
9. S. Cui, Y. Wang, X. Qian and Z. Deng, "Image Processing Techniques in Shockwave Detection and Modeling," *Journal of Signal and Information Processing*, vol. 4, pp. 109-113, 2013.
10. J. Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vols. PAMI-8, no. 6, pp. 679-698, 1986.
11. D. Estruch, N. J. Lawson, D. G. MacManus, K. P. Garry and J. L. Stollery, "Measurement of shock wave unsteadiness using a high-speed schlieren system and digital image processing," *Review of Scientific Instruments*, vol. 79, no. 12, 2008.
12. T. R. Fujimoto, T. Kawasaki and K. Kitamura, "Canny-Edge-Detection/Rankine-Hugoniot-conditions unified shock sensor for inviscid and viscous flows," *Journal of Computational Physics*, vol. 396, pp. 264-279, 2019.
13. M. Monfort, T. Luciani, J. Komperda, B. Ziebart, F. Mashayek and G. E. Marai, "A Deep Learning Approach to Identifying Shock Locations in Turbulent Combustion Tensor Fields," *Modeling, Analysis, and Visualization of Anisotropy*, pp. 375-392, 2017.
14. M. Dehghan Manshadi, H. Vahdat-Nejad, M. Kazemi-Esfeh and M. Alavi, "Speed Detection in Wind-tunnels by Processing Schlieren Images," *IJE TRANSACTIONS A: Basics*, vol. 29, no. 7, pp. 962-967, 2016.
15. B. Colvert, M. Alsalman and E. Kanso, "Classifying vortex wakes using neural networks," *Bioinspiration & Biomimetics*, vol. 13, no. 2, 2018.
16. I. A. Znamenskaya, I. E. Ivanov, I. A. Kryukov and T. A. Kuli-Zade, "Pulsed volume discharge with preionization in two-dimensional gasdynamic flow," *Journal of Experimental and Theoretical Physics*, vol. 95, no. 6, p. 1033-1039, 2002.
17. I. A. Znamenskaya, A. E. Lutskii and I. V. Mursenkova, "The surface energy deposited into gas during initiation of a pulsed plasma sheet discharge," *Technical Physics Letters*, vol. 30, p. 1036-1038, 2004.
18. E. Koroteeva, I. Znamenskaya and I. Doroshchenko, "Experimental and numerical investigation of a flow induced by a pulsed plasma column," *Physics of Fluids*, vol. 30, no. 8, 2018.
19. E. Koroteeva, I. Znamenskaya, D. Orlov and N. Sysoev, "Shock wave interaction with a thermal layer produced by a plasma sheet actuator," *Journal of Physics D: Applied Physics*, vol. 50, no. 8, 2017.
20. I. Sobel, "An Isotropic 3x3 Image Gradient Operator," *Presentation at Stanford A.I. Project 1968*, 2014.
21. A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto and H. Adam, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," arXiv:1704.04861, 2017.