Tomographic Imaging and Deep Learning based Reconstruction of Burner Flames

Bamidele Ogunjumelo, Md. Moinul Hossain, *Senior Member, IEEE*, Gang Lu, *Senior Member, IEEE* and Ali Khan
School of Engineering, University of Kent, Canterbury, Kent CT2 7NT, UK
bao21@kent.ac.uk, m.hossain@kent.ac.uk, g.lu@kent.ac.uk and ak996@kent.ac.uk

Abstract—This paper presents a tomographic and deep learning (DL) technique for the three-dimensional (3-D) reconstruction of burner flames. Two-dimensional (2-D) flame images are obtained using a tomographic imaging system from different directions around the burner. A flame data augmentation technique using a morphological operator is used to generate the complete training and testing datasets. The simultaneous algebraic reconstruction technique (SART) is used to generate the ground truth, i.e., flame cross-sectional datasets. A DL method based on a convolutional neural network (CNN) is employed for the reconstruction of the flame cross- and longitudinal sections. The CNN parameters are optimized through a trial-and-error approach as well as simulation. The CNN is constructed using a machine learning (ML) hardware accelerator i.e., a tensor processing unit to perform faster reconstruction. The proposed model is evaluated using the 2-D flame images obtained on a lab-scale gas-fired test rig under different operation conditions. Results obtained from the experiments suggest that the proposed strategy can accurately and faster reconstruct the flame cross- and longitudinal sections.

Keywords—flame, tomographic imaging, deep learning, 3-D reconstruction

I. INTRODUCTION

Due to increasingly tighter government regulations, the power generation sector and other industries with energy-intensive processes are required to maximize combustion efficiency and reduce pollutant emissions in their combustion systems (such as boilers and gas turbines). Therefore, combustion monitoring and diagnosis play a vital role in controlling and optimizing such combustion processes [1]. A flame, as the central reaction zone of the combustion process, has a number of characteristic parameters such as size, shape, brightness, uniformity, temperature and oscillation frequency [2]. The monitoring and characterization of these parameters have become increasingly important for an in-depth understanding of the combustion process. With the advances in digital imaging and computing technology, digital imaging-based volumetric tomography (VT) techniques have attracted great attention in combustion research due to their unique features including three-dimensional (3-D) visualization, non-intrusiveness, relatively simple system set-up and easy implementation, making them suitable for the spatial and temporal monitoring and characterization on practical combustion systems [3, 4]. Conventional iterative techniques such as the Algebraic Reconstruction Technique (ART) [5] and the Simultaneous ART (SART) [6] were used successfully in the 3-D reconstruction of flames. However, these techniques have limitations such as high computational cost and offline reconstruction.

In recent years, the 3-D reconstruction of burner flames through Deep Learning (DL) has increasingly attracted the interest of combustion researchers and engineers. It has demonstrated an impressive performance in terms of reconstruction accuracy and computational efficiency compared with the conventional 3-D reconstruction approaches [7, 8, 9]. The widely used DL models are the Convolutional Neural Networks (CNNs) which have been applied in various tomographic applications including ultrasonic, magnetic resonance and X-ray computed tomography (CT) [10-12]. For instance, Ying, et al. [7] developed a CNN-based tomography system using 12 color Charge-Coupled Device (CCD) cameras for rapid 3-D flame chemiluminescence reconstruction. Huang, et al. [8] designed a hybrid model utilizing a CNN and Long Short-Term Memory (LSTM) model for reconstructing 3-D flame structures indirectly from 2-D projections without explicitly tomographic reconstruction. Wang, et al. [9] performed the 3-D reconstruction of turbulent flames based on the CNN and recurrent neural network models. To solve the inversion problem in the CT system, Huang, et al. [13] proposed a CNN and proper orthogonal decomposition (POD) based solution for the flame reconstruction. Cai, et al. [14] show the feasibility of using Transfer Learning (TL) for the VT flame reconstruction, where a CNN is implemented using both the TL and semi-supervised learning techniques. The above-mentioned studies demonstrate that the DL techniques can be used for the 3-D reconstruction of flames with a similar level of accuracy as conventional tomographic techniques. However, whilst the DL approaches have proven promising for 3-D flame reconstruction, they rely on a large amount of experimental data, which can sometimes be challenging to obtain.

In addition, the performance of DL models in real-time measurements or online monitoring of burner flames can be improved by machine learning (ML) hardware accelerators. This provides significant performance, energy efficiency, and cost-effectiveness advantages. Hardware accelerators for ML are specifically designed to handle heavy computational demands. Multiple cores or specialized units enable them to perform parallel computations more efficiently than central processing units (CPUs). Using parallelism, larger datasets and more complex models can be trained and inferred faster. In comparison to CPUs, ML hardware accelerators deliver high performance while using less power. Energy efficiency is especially beneficial when ML workloads are deployed in resource-constrained environments, such as mobile devices or edge computing devices. The use of ML hardware accelerators such as tensor processing units (TPUs) [15] is often more cost-effective than scaling a CPU-based infrastructure. ML hardware accelerators can deliver superior performance at a lower price, making them an appealing option for organizations and researchers. Though various DL-based 3-D flame reconstruction applications have been studied, the majority of these have concentrated on CPU and GPU-based implementations.

In this study, a DL-based tomographic imaging technique along with an ML hardware accelerator is proposed to reconstruct 3-D flames cross- and longitudinal sections. A tomographic imaging system is used to capture flame images under different combustion operation conditions. Data augmentation using a morphological transformation operator is further used to increase the data availability for the proposed technique. The ground truth data is generated using the traditional SART. The model construction is performed on a TPU, an ML hardware accelerator, for an accelerated reconstruction process. The flame reconstruction results obtained from the proposed model are discussed and presented.

II. Methodology

Fig. 1 shows the technical strategy of the proposed DL-based 3-D flame reconstruction. A CNN is constructed based on 2-D flame images which are obtained using a tomographic imaging system from different directions around the flame. A SART algorithm is applied to generate ground truth cross-sectional data using the 2-D flame images. The CNN (Fig. 2) consists of seven 2-D convolutional layers (Conv2D), each having a linear activation function. This is chosen based on a trial-and-error method for achieving the best performance of the model. The output layer has a flattened layer and a dense layer. Other hyperparameters such as learning rate and batch size are determined through simulations. The CNN is trained using the 2-D flame images and the ground truth data. The trained model is then applied to reconstruct the flame's cross- and longitudinal sections.

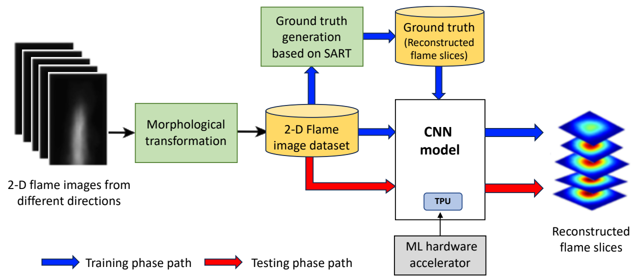


Fig. 1. Technical strategy of the proposed DL-based 3-D flame reconstruction.

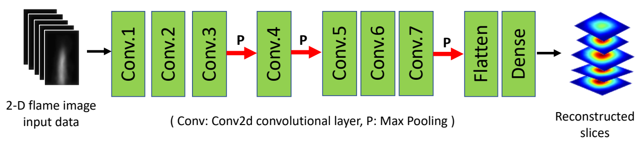


Fig. 2. The architecture of the CNN model.

In this study, the CNN is constructed based on the Conv2D layers using Keras API (application programming interface) [16]. The Conv2D layer determines features of the input image through a kernel and produces an output feature map [i.e., indicates whether the learned features are presented in the input image]. This procedure can be repeated by applying multiple kernels to produce various output feature maps where each feature map represents a different characteristic of the input image. The Conv2D layer uses important hyperparameters such as the number of kernels, kernel size and stride and these parameters can be tuned during the CNN training process.

In the Keras API, the first hyperparameter to be set for the Conv2D layer is the number of convolution kernels. Each kernel represents a different set of features in the output feature map. In this study, the first Conv2D layer has the number of kernels of 8; the second and third Conv2D layers, 16; the fourth and fifth Conv2D layers, 32; and the last two Conv2D layers (i.e., sixth and seventh), 64. The kernel size specifies the height and width of each Conv2D window. The common kernel sizes are (1×1), (3×3), (5×5) and (7×7). In this study, the kernel size of (3×3) is chosen as it is considered the best practice in the field [17]. The stride specifies the step size of the convolution taken along the x and y axes of the input data. The stride value is set to (1, 1) to ensure that the convolutional kernel is moved one pixel at a time from the left to the right of the input data. This process is repeated until the convolutional kernel reaches the far-right border of the data. The kernel is then shifted to one pixel down and re-restart the process. In this way, the convolution process is completed for the whole input data. Padding is another important parameter which has two types, i.e., valid and same for the Conv2D layers. The "same" type is used in this study to ensure that padding is applied in such a way that the output feature map has the same spatial dimensions as the input image. This also ensures that the kernel is applied to all the pixels of the input images. In addition, three max-pooling layers are employed between the Conv2D layers to reduce the spatial size of the feature map.

The CNN was trained on Google Colab [18] on a TPU. The Google Colab is a cloud computing platform, that can access computing resources such as CPUs, GPUs and TPUs, allowing developers to write and execute Python code through a web browser by using a hosted Jupyter Notebook service. The TPU is Google's custom-developed Application Specific Integrated Circuit (ASIC) which uses a math library called TensorFlow framework, enabling it to accelerate machine learning workloads, particularly CNN modelling. A detailed description of the configuration and setup of the TPU can be found elsewhere [15] [19].

III. Data Preparation and Model Construction

A. Data Preparation

2-D images of the flame were obtained using a 3-D flame tomographic imaging system from eight different directions around a gas-fired Bunsen-type burner under six different operation conditions. A detailed description of the imaging system can be found in [20]. Fig. 3 shows the example images of the flame. The dimensions of each image projection dataset are (8×145×90) where 8 is the number of projections (145×90) and the size of each projection (i.e., 2-D flame image). For each operation condition, a total of 100 samples were used for training and testing the CNN, where ten samples were original, i.e., without altering, and 90 samples were generated using the data augmentation technique as described in the next section.

B. Data Augmentation

Due to the data availability and accessibility, additional flame projection datasets need to be generated to train the CNN effectively. This is achieved through a morphological transformation over the ten original 2-D flame images. These samples were then added to the original 2-D images to ensure that the dataset contained images with different shapes and structures.

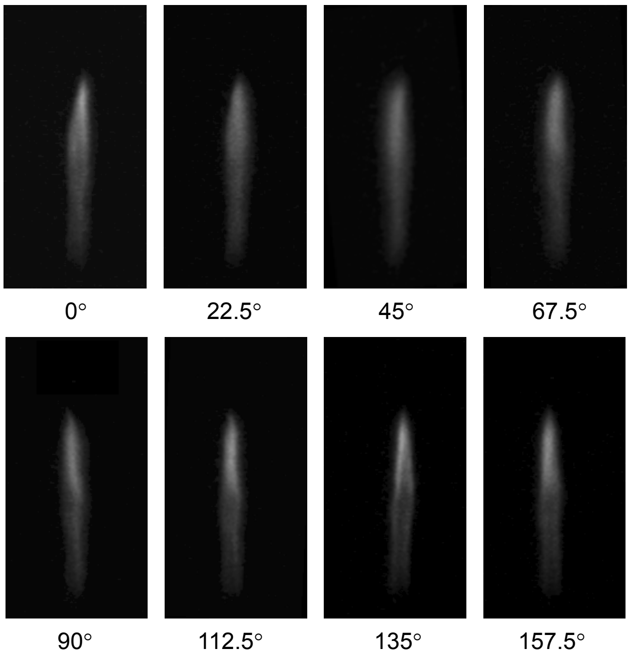


Fig. 3 Example flame images taken from eight different directions using the tomographic imaging system [20].

TABLE I. FLAME SAMPLES USED IN DATASET PREPARATION

Condition	Fuel flow rate (l/min)	No. of original images	No. of transformed images	Total number of images
1	0.2	10	90	100
2	0.3	10	90	100
3	0.4	10	90	100
4	0.5	10	90	100
5	0.6	10	90	100
6	0.7	10	90	100

The morphological transformation operator essentially changes the geometric and luminous appearance of the original flame images without losing their main characteristics. To obtain a wide range of images, random values for kernel and iteration were used in the transformation. The output of the operation is a set of projection images which are similar to the original input images. Fig. 4 shows the samples of transformed flame images. These images are similar in terms of their structural similarity index measure (SSIM) values (Image A: 0.92, Image B: 0.62 and Image C: 0.83). Using this approach, a total of 100 projection samples (which includes ten original flame samples) were generated for each condition as shown in Table I, making a total of 600 samples available for training, testing and validating the CNN (Fig. 2).

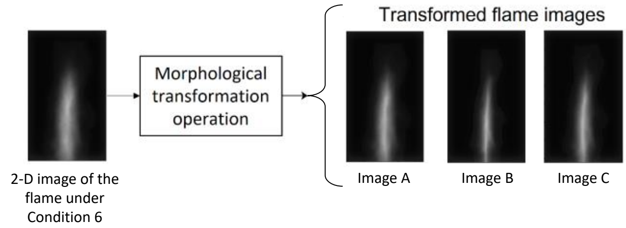


Fig.4 Data augmentation for additional dataset generation.

C. Ground Truth

A total of 600 ground truth data were generated, which are a collection of cross-sectional slices of the reconstructed flame using the SART. Fig. 5 illustrates typical cross-sectional slices generated for Condition 6. The dimension of each ground truth dataset is (145, 90, 90), where 145 is the number of slices and (90, 90) is the dimensions of each 2-D image.

D. Model Construction

To construct the CNN, the dataset was split into training (60%), validation (20%), and testing (20%) sub-datasets. Before the training, the training dataset was shuffled to mix up the samples and the 2-D projection data. The dimensions of the training dataset were reduced to (8×25×25) and the ground truth data to (145×45×40). This was due to the limited availability of computation resources. The model is initially trained using different values (i.e., the number of convolutional layers, learning rate and batch size) and iterations to determine the optimum hyperparameters. The model was implemented in Python using Keras under the TensorFlow framework. It was trained using 1000 epochs as the training and validation losses did not reduce further after this value (i.e., the model reached a steady state). A learning rate of 0.001 and a batch size of 16 were used during the training process. It took two hours to train the model on the TPU ML hardware accelerator.

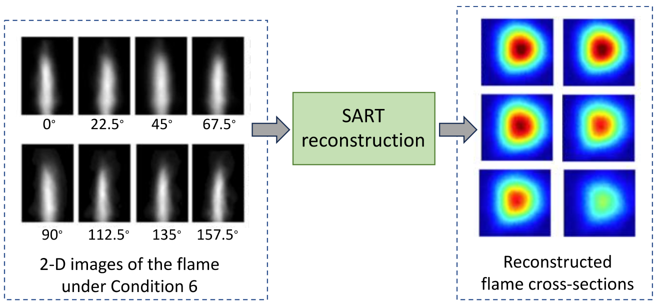


Fig.5 Example of the ground truth (i.e., flame cross-sectional data) generated using the SART.

IV. Results and Discussion

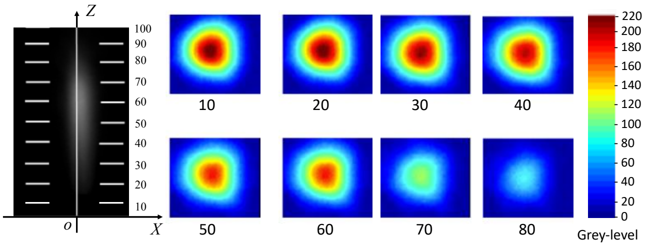
A. 3-D Reconstruction of Flame Sections

Fig. 6 illustrates the grey-level reconstruction of the flame cross- and longitudinal sections achieved using the proposed model for Condition 6 (Table I). To evaluate the quality of the reconstruction, a total of 145 slices of flame were used under Condition 6. An average SSIM was found to be 0.77±0.07, which suggests that the ground truth images and reconstructed slices have a high degree of similarity. An average Root Mean Squared Error (RMSE) is also found to be 9.7e-04. In addition, the peak signal-to-noise ratio (PSNR) value is 60.9 ± 2.87, suggesting that the reconstruction results have high accuracy and low variability across the sample sets. It is demonstrated that the proposed DL model can perform the online reconstruction of the cross- and longitudinal sections of the flames under various combustion conditions.

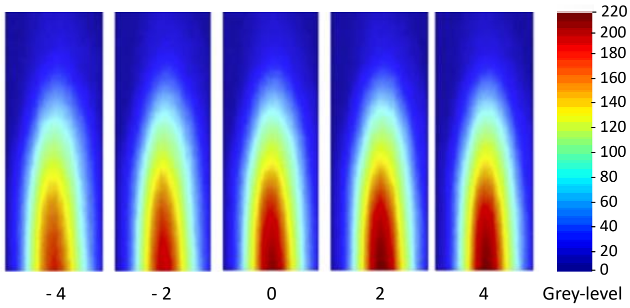
B. *Computational Efficiency*

To investigate the computational efficiency, the flame reconstruction as given in section IV was performed using both the TPU and CPU on the Colab platform. The flame reconstruction using other models was also conducted. The computational times of the reconstruction are given in Table II. It can be seen that the reconstruction time using the TPU is 77% faster than that of the CPU.

It is also found that the proposed strategy has faster reconstruction compared with the conventional Multiplicative ART (MART) and ART on a CPU-based platform [7]. However, it is difficult to compare with the results from other studies since the flame structure, the hardware platform, code optimization, and model architecture differ from one to another. Further research is needed for fair and effective comparison among different models on different platforms.



(a) Cross-sections at different flame heights (along the *z*-axis) as shown in the flame image at left. Note: the cross-section numbers are arbitrary.



(b) Longitudinal sections at different flame depths (along the *x*-axis). 0: the longitudinal section at the burner axis; -2 and -4: longitudinal sections back-off the burner axis; 2 and 4: sections forward the burner axis.

Fig. 6 Grey-level distributions of flame cross- and longitudinal-sections.

TABLE II. COMPARISON OF FLAME RECONSTRUCTION TIME

Reconstruction method	**Hardware platform**	**Reconstruction time (s)**
CNN [Proposed]	TPU	0.24
	CPU [Intel(R) Xeon (R), 2.30 GHz]	1.07
CNN [7]	CPU [Intel Core i7-8750H, 2.20 GHz]	1.28
MART [7]		143.40
ART [7]		170.25

V. CONCLUSION

A deep learning and tomographic imaging-based technique incorporated with a machine learning hardware accelerator has been proposed for the 3-D reconstruction of burner flames. 2-D images of the gas-fired flame from eight different directions around the burner are obtained using a 3-D flame tomography system under six different operation conditions. A morphological transformation has been performed to generate complete flame datasets based on the original 2-D flame images. A convolutional neural network has been constructed, where the 2-D flame images are used as input, and flames cross- and longitudinal sections derived from the SART algorithm serve as ground truth data. The experimental results have demonstrated that reconstructed flame slices (i.e., cross- and longitudinal sections) and the ground truth data have high similarity. A comparative study has shown that the proposed CNN has a faster reconstruction time compared with other iterative methods. Further work will focus on experimenting with more flame conditions and effectively comparing reconstruction times observed in other studies.

REFERENCES

[1] Y. Yan, T. Qiu, G. Lu, M. M. Hossain and G. Gilabert, "Recent advances in 3D flame tomography", Proc. 6th World Congr. Ind. Process Tomogr, pp. 1530-1539, 2010.

[2] Y. Yan, G. Lu and M. Colechin, "Monitoring and characterisation of pulverised coal flames using digital imaging techniques", Fuel, vol. 81, pp. 647-655, 2002.

[3] B. R. Halls, P. S. Hsu, S. Roy, T. R. Meyer and J. R. Gord, "Two-color volumetric laser-induced fluorescence for 3D OH and temperature fields in turbulent reacting flows," Opt. Lett. 43, pp. 2961-2964, 2018.

[4] L. Hecong, S. Bin and C. Weiwei, "kHz-rate volumetric flame imaging using a single camera", Optics Communications, vol. 437, pp. 33-43, 2019.

[5] R. Gordon, R. Bender and G. T. Herman, "Algebraic reconstruction techniques (ART) for 3-dimensional electron microscopy and X-Ray photography", Journal of Theoretical Biology, vol. 29, pp. 471, 1970.

[6] A. H. Andersen and A. C. Kak, "Simultaneous algebraic reconstruction technique (SART) - A superior Implementation of the ART algorithm", Ultrasonic Imaging, vol. 6, pp. 81-94, 1984.

[7] Y. Jin, W. Zhang, Y. Song, X. Qu, Z. Li, Y. Ji and A. He, "Three-dimensional rapid flame chemiluminescence tomography via deep learning," Opt. Express 27, 27308-27334, 2019.

[8] J. Huang, H. Liu and W. Cai, "Online in situ prediction of 3-D flame evolution from its history 2-D projections via deep learning", Journal of Fluid Mechanics, vol. 875, pp. R2-1-12, March 2019.

[9] J. Huang, H. Liu, Q. Wang and W. Cai, "Limited-projection volumetric tomography for time-resolved turbulent combustion diagnostics via deep learning", Aerospace Science and Technology, vol.106 (206123), Nov. 2020.

[10] T. Lähivaara, L. Kärkkäinen, J. M. J. Huttunen and J. S. Hesthaven, "Deep convolutional neural networks for estimating porous material parameters with ultrasound tomography, The Journal of the Acoustical Society of America, vol. 143, pp. 1148-1158, 2018.

[11] S. Wang, Z. Su, L.Ying, X. Peng, S. Zhu, F. Liang, D. Feng and D. Liang, "Accelerating magnetic resonance imaging via deep learning", Proc. 13th IEEE Intl. Symp. Biomed Imaging, pp. 514-517, Prague, Czech Republic, 13-16 April 2016.

[12] H. Chen, Y. Zhang, W. Zhang, P. Liao, K. Li, J. Zhou and G. Wang, "Low-dose CT via convolutional neural network", Biomedical Optics Express, vol.8, pp.679-694, 2017.

[13] J. Huang, H. Liu and W. Cai, "Tomographic reconstruction for 3D flame imaging using convolutional neural networks", Proc. 12th Asia-Pacific Conference on Combustion, Fukuoka, Japan, 1-5 July 2019.

[14] W. Cai, J. Huang, A. Deng and Q.Wang, " Volumetric reconstruction for combustion diagnostics via transfer learning and semi-supervised learning with limited labels", Aerospace Science and Technology, vol. 100, March 2021.

[15] J. Zuern, "Using a TPU in Google Colab", https://jannik-zuern.medium.com/using-a-tpu-in-google-colab-54257328d7da (last access: 20/05/2023).

[16] A. Gulli, A. Kapoor and S. Pal, "Deep learning with TensorFlow 2 and Keras" (2nd edition), Packt Publishing, ISBN-10: 1838823417, 2019.

[17] I. Zafar, G. Tzanidou, R. Burton, N. Patel and L. Araujo, "Hands-on convolutional neural networks with TensorFlow", Packt Publishing, ISBN-10: 1789130336, Aug. 2018.

[18] T. Carneiro, R. V. Medeiros Da NóBrega, T. Nepomuceno, G. B. Bian, V. H. C. De Albuquerque and P. P. R. Filho, "Performance analysis of Google Colaboratory as a tool for accelerating deep learning applications," IEEE Access, vol. 6, pp. 61677-61685, 2018.

[19] A. Shahid and M. Mushtaq, "A survey comparing specialized hardware and evolution in TPUs for neural networks", Proc. 2020 IEEE 23rd International Multitopic Conference (INMIC), Bahawalpur, Pakistan, pp. 1-6, 2020

[20] M. M. Hossain, G. Lu and Y. Yan, "Optical fiber imaging based tomographic reconstruction of burner flames," IEEE Transactions on Instrumentation and Measurement, vol. 61, pp. 1417-1425, 2012.