Flare Gas Flow Rate Estimation Using Multimodal Deep Learning

Yu Watanabe[1], Kento Ishii[2], Nana Tamai[3], Takehisa Yairi [4], and Naoya Takeishi[5]

[1,4,5] *The University of Tokyo, Meguro, Tokyo, 153-8904, Japan*

watanabe-yuu31724@g.ecc.u-tokyo.ac.jp
yairi@g.ecc.u-tokyo.ac.jp
ntake@g.ecc.u-tokyo.ac.jp

[2,3]*ENEOS Holdings, Inc, Yokohama, Kanagawa, 231-0815, Japan*

ishii.kento@eneos.com
tamai.nana@eneos.com

ABSTRACT

In refinery operations, flare gas is generated as a byproduct. It is not only harmful to human health and the environment but also causes secondary issues such as noise and unpleasant odors. Flare stacks are commonly used to combust and neutralize flare gas before releasing it into the atmosphere. While accurate monitoring of flare gas flow rate is essential for flare gas reduction and recovery, installing flow meters on existing facilities can be costly. This study proposes a method to estimate flare gas flow rate using the flare images and suppression steam flow rate. Flare images are processed with a convolutional neural network (CNN) to extract spatial features, while suppression steam time-series data are processed with a long short-term memory (LSTM) network to capture temporal dynamics. These features are fused and passed through fully connected layers to regress the flare gas flow rate. To address data imbalance due to the infrequent occurrence of flare events, we designed a custom loss function that assigns higher weights to high-flow samples while penalizing overestimation when low-flow samples are incorrectly predicted as high flow. Furthermore, we employed data augmentation, preprocessing techniques, and feature engineering to improve prediction accuracy.

1. INTRODUCTION

During the operation of refineries and chemical plants, flammable flare gas is generated as a byproduct. These gases contain hazardous substances such as SOx, NOx, VOCs, and CO, which can lead to ozone layer depletion, acid rain, and potential health hazards. In addition, flare activity may cause light pollution at night, noise, and unpleasant odors, which pose environmental concerns, particularly in refineries located near residential areas. Such flare gases are combusted and detoxified using a facility called a flare stack, which releases the resulting emissions into the atmosphere. A flare stack typically consists of a main burner for combustion, a pilot burner to maintain continuous ignition, and a steam supply system that suppresses black smoke generation. Given the aforementioned environmental and social concerns, accurate monitoring of flare activity is essential to promote emission reduction and gas reuse.

While real-time and accurate measurement of flare gas flow is highly effective for reducing flare emissions and mitigating environmental impact, installing flow meters on-site is costly. Moreover, such meters may exhibit reduced accuracy due to variations in gas composition and temperature. They must also cover a wide measurement range to capture rare but large flare events, which makes it difficult to detect fine-grained variations during normal operation. Currently, most monitoring is conducted via visual inspection.

This study aims to improve the accuracy and reliability of flare monitoring in real-world settings through a practical and deployable approach, leveraging existing visual and sensor data without the need for additional hardware investment. We propose a deep learning-based method for estimating flare gas flow using flare images and time-series data of steam flow. Spatial features are extracted from the images using a convolutional neural network (CNN; LeCun et al., 1998), and temporal trends in the steam flow are captured using a long short-term memory network (LSTM; Hochreiter & Schmidhuber, 1997). These representations are then integrated to construct a regression model for flow prediction.

Since large flare events are abnormal and rare in well-managed facilities, the available training data for high-flow conditions is extremely limited. This results in an imbalanced data distribution, which is known to cause bias and reduced

generalization in deep learning models (Johnson & Khoshgoftaar, 2019). To mitigate this issue, we apply a weighted loss function that assigns greater importance to the small number of high-flow samples. In addition, we introduce penalties for overestimating low-flow samples (and *not* for underestimating high-flow samples), helping to suppress false positives. This approach enhances the model's robustness against data imbalance.

Furthermore, to improve input diversity, we apply basic image transformations such as brightness and contrast adjustments for data augmentation. Even simple techniques of this kind have been shown to be effective for improving the performance of deep learning models (Perez & Wang, 2017).

2. FLARE GAS FLOW RATE ESTIMATION

2.1. Task

The task is to estimate gas flow rate using steam flow rates and flare images. In general, a higher flare gas flow rate is associated with a larger flame area. However, the appearance of the flame is also affected by the amount of steam injected to suppress black smoke generation. Consequently, information on the steam flow rate is essential for estimating the flare gas flow rate. Furthermore, the appearance of the flare is greatly influenced by day and night differences and weather conditions, which makes the accurate estimation of the flare gas flow rate more challenging.

We develop a method for such a task and demonstrate its effectiveness with real-world plant operation data collected from an actual refinery.

2.2. Data Acquisition

In this study, we used data from a flare stack facility equipped with a flare gas flow meter. The measured gas flow values served as ground truth labels in the training data and were used to evaluate prediction errors in the test data.

Gas flow rate and steam flow rate were recorded at 1-minute intervals, while flare images were captured at 10-minute intervals. Consequently, timesteps at 10-minute intervals contain synchronized data across all three modalities.

2.3. Data Characteristics

Our dataset comprises three distinct periods that include high gas flow events. For each period, Table 1 summarizes the date range, total number of samples at 10-min intervals, and the number and percentage of samples exceeding a heuristic threshold set to detect high-flow events. Notably, the dataset exhibits a significant class imbalance: low-gas flow samples are predominant, while high-gas flow samples are rare.

To preserve confidentiality, in this paper, the threshold is denoted just by τ without revealing its concrete value, and all flow values are normalized by dividing them by the global maximum value observed across all periods.

Table 1. Overview of dataset

Period	Date Range	Total number of Samples at 10-min intervals	Number of Samples Above Threshold τ (Percentage)
A	Oct 25-Nov 30, 2023	5235	58 (1.11 %)
B	Mar 1-Mar 31, 2024	4464	169 (3.79 %)
C	May 1-Jun 10, 2024	5904	227 (3.84 %)

2.4. Data Overview

Figure 1 and Figure 2 respectively illustrate the gas flow rate and steam flow rate over time for each period (A, B, and C). Figure 3 shows representative flare images captured under different gas flow conditions and at various times of day (e.g., morning and night). These images reveal visual differences in flare appearance influenced not only by flow rate but also by environmental conditions such as lighting and cloud coverage.

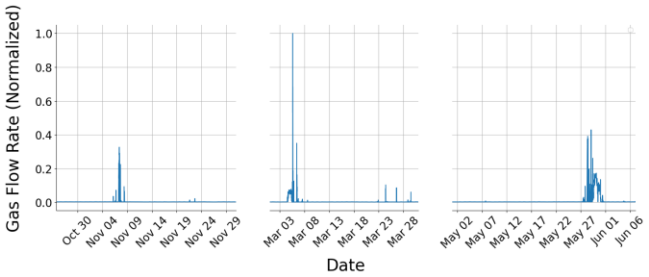

Figure 1. Normalized Gas Flow Rate Over Time

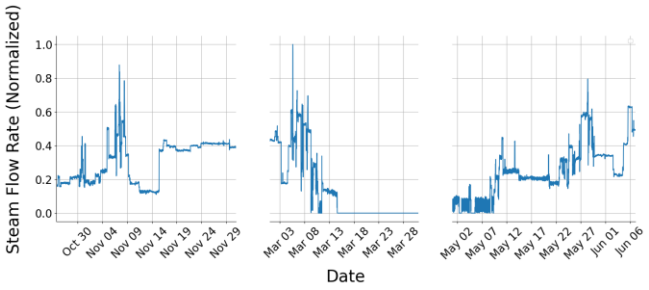

Figure 2. Normalized Steam Flow Rate Over Time

Figure 3. Examples of Flare Images under Varying Flow and Environmental Conditions

3. METHODS

We use a multimodal learning approach for estimating gas flow rate using flare images and steam flow rate. The task is formulated as a regression problem, where the target variable is the gas flow rate at each time step.

3.1. Model Architectures

We design and compare three neural network models that share a common convolutional neural network (CNN) encoder for processing flare images. The key difference among the models lies in how steam flow rate data is processed. Figures 4–6 illustrate the architectures of the three models.

3.1.1. Flare Image

Input RGB images of size 128×128 pixels are processed through two convolutional layers with 3×3 kernels and padding of 1. The number of channels is increased from 3 to 16 in the first layer and from 16 to 32 in the second layer. Each convolutional layer is followed by a ReLU activation and a 2×2 max-pooling layer that reduces spatial dimensions by half. After the second pooling, the resulting 32 feature maps have spatial dimensions of 32×32. These feature maps are flattened and passed through a fully connected layer to produce a 32-dimensional feature vector.

3.1.2. Steam Flow Rate

For processing steam flow rate information, we try three different models as follows:

- Model 1 (Single Steam Flow Rate): This model uses only the steam flow rate at the same time step as the flare image. The input steam flow rate value is simply concatenated to the output feature vector obtained from the CNN processing of the flare image.
- Model 2 (Sequential Steam Flow Rate): This model incorporates a sequence of past steam flow rates, including time steps where no image is available. The sequence, consisting of 10 time steps sampled at 1-minute intervals, is processed using a long short-term memory (LSTM) network to capture temporal dependencies. The LSTM has a single layer with a hidden size of 32, and the final hidden state is used as the steam flow feature representation.
- Model 3 (Sequential and Differential Steam Flow Rate): This model incorporates a sequence of past steam flow rates and corresponding first-order difference sequence. The difference sequence has 9 time steps. Both sequences include time steps where no image is available. Each sequence is processed by a separate LSTM encoder with the same architecture as Model 2.

3.1.3. Fusion and Prediction

After encoding each modality, the resulting features are concatenated and passed to a linear output layer for gas flow rate prediction.

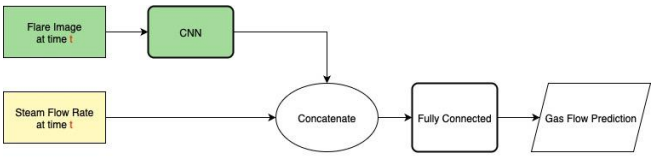

Figure 4. Model 1 (Single Steam Flow Rate)

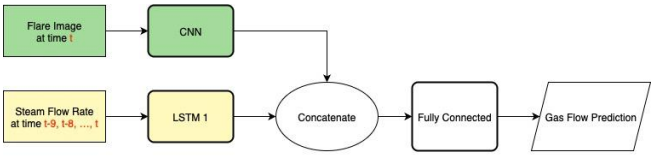

Figure 5. Model 2 (Sequential Steam Flow Rate)

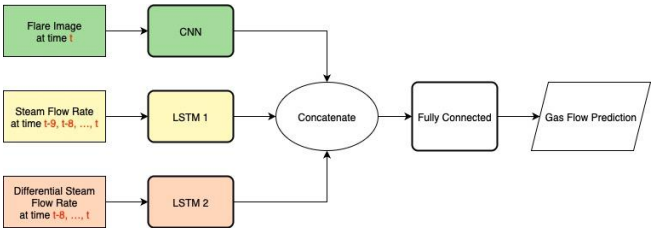

Figure 6. Model 3 (Sequential and Differential Steam Flow Rate)

3.2. Loss Function

To address the asymmetry in importance between high-flow and low-flow predictions, we introduce a custom Wighted Mean Squared Error (Weighted MSE) loss function. The objective of the loss function is to prioritize accurate prediction of high gas flow events, as failing to detect these critical conditions can have significant consequences. Simultaneously, the loss penalizes overestimation during low-flow periods to reduce false positive alarms.

Let y_i denote the true gas flow rate and $\hat{y}_i$ the predicted value at sample i. The weighted MSE is defined as:

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} w_i(\hat{y}_i - y_i)^2 \quad (1)$$

where w_i is a sample-specific weight determined by:

$$w_i = \begin{cases} w_{high}, & if\ y_i > \tau \\ w_{low}, & if\ y_i \le \tau\ and\ \hat{y}_i > \tau \\ 1, & otherwise \end{cases} \quad (2)$$

where τ is the threshold defined in Section 2.2 to identify high-flow events.

3.3. Data Augmentation

To increase the size and diversity of the training dataset, we generate additional samples by applying image processing operations such as color jittering, which randomly adjusts brightness, contrast, saturation, and hue, to the original images. This approach can double the dataset size and enhance data variety. Figure 7 shows pairs of original images and their augmented images produced by applying color jittering.

Figure 7. Examples of original and augmented image pairs (left: original images, right: augmented images)

4. EXPERIMENTS

We conducted experiments using the dataset described in Section 2 to evaluate the effectiveness of the proposed models.

4.1. Experimental Setup

We used data from periods A and B for training, and data from period C for testing.

For image preprocessing, the right half of each image was extracted to focus on the region of interest corresponding to the target flare stack. The cropped images were subsequently resized such that their shorter side measured 128 pixels, followed by a center crop to obtain a fixed resolution of 128×128 pixels.

All experiments were repeated three times, and the reported results represent the average performance across these runs.

4.2. Evaluation Metrics

To evaluate the model performance, we use two metrics, high-flow MSE, and low-flow MSE. High-flow MSE specifically measures the prediction error on samples where the true gas flow rate exceeds a predefined threshold τ. Low-flow MSE measures the error on samples below the threshold τ.

4.3. Results

4.3.1. Steam Flow Rate

We evaluated the performance of the three models described in Section 3.1.2, which differ in how they incorporate steam flow rate information. Model 1 uses a single steam flow value. Model 2 uses a temporal sequence of past steam flow rates, and Model 3 uses both the original steam flow rate sequence and its first-order difference. For this experiment, we used MSE loss (i.e., $w_{high} = w_{low} = 1$).

Table 2 represents a comparison of the average model performance in terms of high-flow MSE, low-flow MSE for the three different input structures of the steam flow rate. The results indicate that incorporating temporal sequences (Model 2) and additional differential temporal sequences (Model 3) leads to a reduction in high-MSE, while maintaining comparable low-flow MSE performance.

Figures 8, 9, and 10 show the time series of predicted gas flow rates obtained using Models 1, 2, and 3, respectively. Each figure compares the predicted values with the ground truth values over the evaluation period. In Model 1, the high-flow events are not well captured, with most predictions tending to classify the flow as low-flow. In contrast to Model 1, Models 2 and 3 successfully capture the high-flow events, with Model 3 providing predictions that more closely match the ground truth values.

Table 2. Comparison of Average Model Performance based on Steam Flow Rate Input Structures

Model	High-flow MSE	Low-flow MSE
Model 1 (Single)	148.6	0.320
Model 2 (Seq.)	128.2	0.372
Model 3 (Seq.+Diff)	122.5	0.206

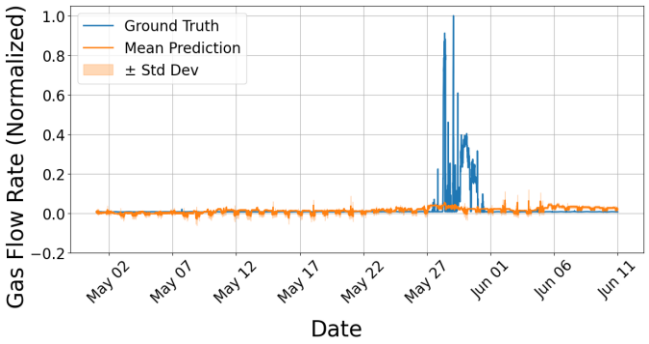

Figure 8. Gas Flow Rate Prediction Using Model 1

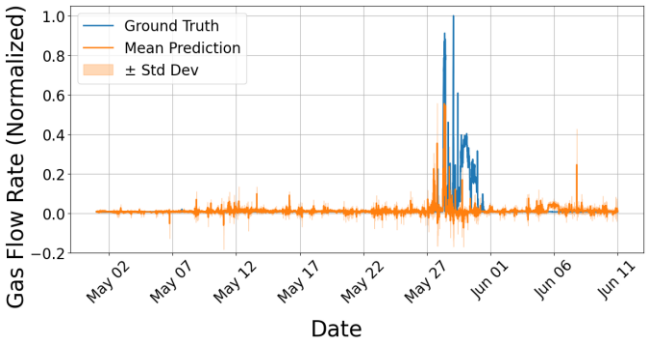

Figure 9. Gas Flow Rate Prediction Using Model 2

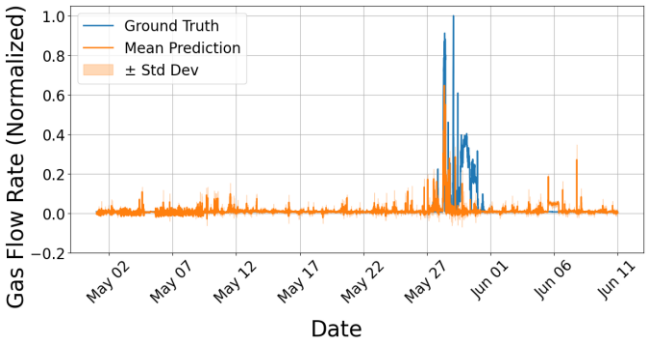

Figure 10. Gas Flow Rate Prediction Using Model 3

4.3.2. Loss Function

We experimented with different combinations of w_{high} and w_{low}. w_{high} is the weight applied to high-flow samples, emphasizing their importance in the loss function. w_{low} serves as a penalty when the model incorrectly predicts a high gas flow rate during low-flow periods. We evaluated 9 combinations of loss weights, with $w_{high} \in \{1, 5, 10\}$ and $w_{low} \in \{1, 3, 5\}$. For this experiment, we used Model 3, which showed the best performance among the steam flow rate input structures in Section 4.3.1.

Table 3 represents a comparison of the average model performance in terms of high-flow MSE, low-flow MSE under different loss weight configurations. The results show that increasing w_{high} leads to a reduction in high-flow MSE, indicating improved accuracy for high-flow events. However, this is accompanied by an increase in low-flow MSE. This trade-off highlights the need to carefully balance loss weights based on the desired performance priorities.

Figure 11 and Figure 12 respectively show the time series of predicted gas flow rate with the lowest high-flow MSE and the lowest low-flow MSE.

Table 3. Comparison of Average Model Performance based on Loss Weight

w_{high}	w_{low}	High-Flow MSE	Low-Flow MSE
1	1	122.49	0.206
5		94.24	2.555
10		89.60	8.044
1	3	130.18	0.175
5		119.57	1.143
10		131.66	1.742
1	5	125.10	0.167
5		142.34	1.029
10		120.65	0.954

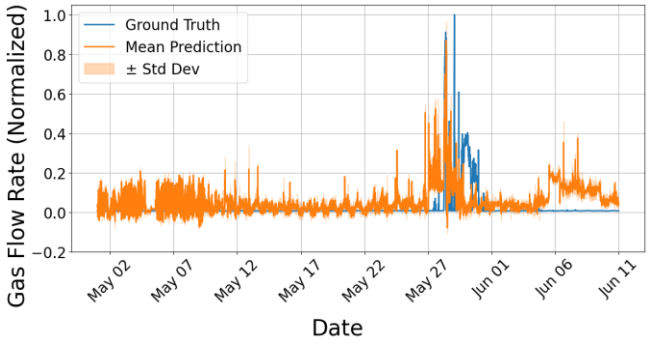

Figure 11. Gas Flow Rate Prediction Using Loss Weights $w_{high} = 10, w_{low} = 1$

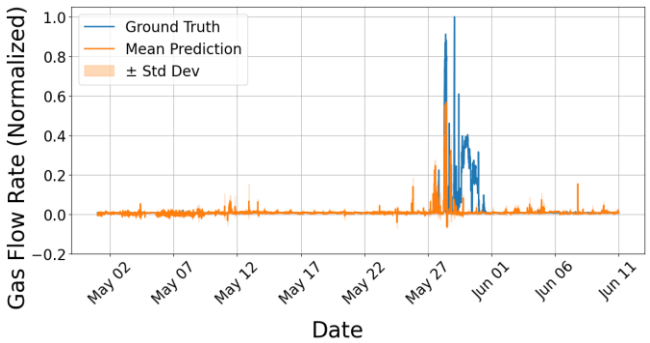

Figure 12. Gas Flow Rate Prediction Using Loss Weights $w_{high} = 1, w_{low} = 5$

4.3.3. Data Augmentation

We evaluated the performance with data augmentation. For this experiment, we used Model 3, which showed the best performance among the steam flow rate input structures in Section 4.3.1. The evaluation was conducted for 9 combinations of $w_{high} \in \{1, 5, 10\}$ and $w_{low} \in \{1, 3, 5\}$.

Table 4 represents a comparison of the average model performance in terms of high-flow MSE, low-flow MSE with and without data augmentation. The results show that data augmentation tends to reduce high-flow MSE in many configurations.

Figure 13 shows the time series of predicted gas flow rate with data augmentation ($w_{high} = w_{low} = 1$). Compared to Figure 10, which shows the result without augmentation, the predictions in Figure 13 during high-flow events are more accurate.

Table 4. Comparison of Average Model Performance With and Without Date Augmentation

w_{high}	w_{low}	Without Augmentation		With Augmentation	
		High-flow MSE	Low-flow MSE	High-flow MSE	Low-flow MSE
1	1	122.49	0.206	115.74	0.328
5		94.24	2.555	92.89	2.666
10		89.60	8.044	103.43	1.820
1	3	130.18	0.175	130.68	0.255
5		119.57	1.143	110.75	0.970
10		131.66	1.742	101.91	1.226
1	5	125.10	0.167	141.57	0.158
5		142.34	1.029	110.87	0.639
10		120.65	0.954	112.09	1.183

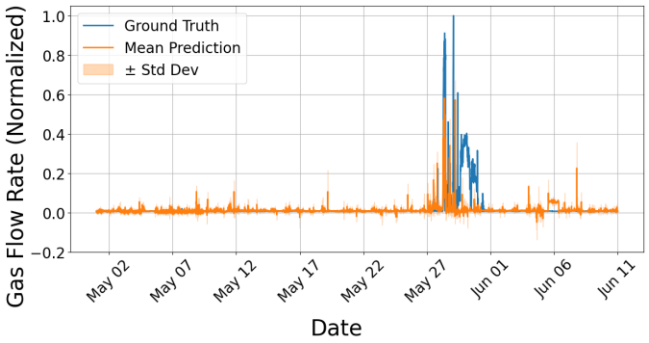

Figure 13. Gas Flow Rate Prediction With Data Augmentation
$w_{high} = 1, w_{low} = 1$

5. CONCLUSION

We have constructed a multimodal neural network that processes the time series of steam flow rates and their first-order differences using separate LSTM networks, processes flare images using a CNN, and integrates these modalities for gas flow rate prediction. Compared to models that use only single steam flow rate values or incorporate a sequence of past steam flow rates, the model that additionally incorporates the first-order difference sequence was more effective in capturing high-flow events and provided predictions that more closely matched the ground truth.

Furthermore, to address the data imbalance caused by the limited number of high-flow samples, we have also designed a customized loss function that assigns higher importance to high-flow predictions and penalizes overestimation in low-flow cases. We observed that placing more emphasis on high-flow events in the loss function tended to improve accuracy in predicting high-flow events, while increasing the penalty for overestimating low-flow events helped reduce errors in such cases. These findings suggest that adjusting the loss weights allows flexible control over prediction accuracy for different flow rate conditions.

To further enhance model performance, we also applied data augmentation to the flare images, generating additional training samples by introducing variations in brightness, contrast, saturation, and hue. We observed an improvement in high-flow MSE, indicating that data augmentation helps improve prediction accuracy.

REFERENCES

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278-2324. doi:10.1109/5.726791

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, vol. 9, no. 8, pp. 1735-1780. doi:10.1162/neco.1997.9.8.1735

Johnson, J. M., & Khoshgoftaar, T. M. (2019). Survey on deep learning with class imbalance. *Journal of Big Data*, vol. 6, no. 1, pp. 1-54. doi:10.1186/s40537-019-0192-5

Perez, L., & Wang, J. (2017). The effectiveness of data augmentation in image classification using deep learning, *arXiv preprint*, arXiv:1712.04621.