

Gear Damage Estimation Based on Impulse Accumulation Model and Feedback-Guided Domain Adaptation

Yoshiki Kato^{1,2}, Koji Wakimoto³, Takahiko Masuzaki⁴, Nobuaki Tanaka⁵, Toshiyuki Kuriyama⁶ and Hanqi Su⁷

^{1,3,4,5,6}*Mitsubishi Electric Corporation, Kamakura, Kanagawa, 247-8501, Japan*

Kato.Yoshiki@dc.MitsubishiElectric.co.jp

Wakimoto.Koji@df.MitsubishiElectric.co.jp

Masuzaki.Takahiko@dc.MitsubishiElectric.co.jp

Tanaka.Nobuaki@ce.MitsubishiElectric.co.jp

Kuriyama.Toshiyuki@dx.MitsubishiElectric.co.jp

⁷*Center for Industrial Artificial Intelligence, Department of Mechanical Engineering, University of Maryland, College Park, MD, 20742, USA*

hanqisu@umd.edu

²*Division of Electrical, Information and Communication Engineering, Graduate School of Natural Science and Technology, Kanazawa University, Kanazawa, Japan*

1. INTRODUCTION

This paper presents a gear damage estimation method developed through our participation in the PHM 2026 Data Challenge.

The main challenges in this competition are described below.

- (1) **Lifetime damage estimation:** Participants are required not only to detect the occurrence of an anomaly but also to quantify damage progression over the entire lifecycle. Various condition indicators have been developed to detect changes in gears (Sharma & Parey, 2016). However, no established indicator has been shown to accurately capture damage progression throughout an entire lifecycle.
- (2) **Lack of ground-truth labels:** The datasets provided for this competition do not include ground-truth labels for the target metric. Participants must create ground-truth labels based on the image data included in the training dataset, train a model, and then apply the model to the test and validation datasets. Their estimation accuracy is then evaluated by the organizers. Since accuracy is evaluated using ground-truth labels withheld by the organizers, participants are required to train a model that predicts the target metric without access to the actual labels. In real-world applications, models developed using data from one set of experiments often need to be

adapted to data from other experiments conducted under different conditions, despite having few or no ground-truth labels for the target data. This competition provides a realistic simulation of such a challenging situation.

To address these challenges, we attempted the following approaches.

- (1) **Impulse Accumulation Model and Condition Indicator Ensemble:** Various indicators are known for detecting damage in gear condition monitoring (Sharma & Parey, 2016; Su & Lee, 2024a; Su & Lee, 2024b; Lee & Su, 2025), and some of those indicators are included in the datasets provided in this competition. However, we could not identify any indicator that directly quantifies damage progression based solely on the indicator at a given point in the lifecycle, especially during the early stages. Therefore, we introduced a model that estimates the damage value based on the cumulative sum of the impulse magnitude sensed during the operation.

We first estimated damage progression during the early stages using the impulse accumulation model. After the damage had progressed beyond a certain level, we applied an ensemble approach that estimated damage progression from vibration data at that point using established condition indicators.

- (2) **Feedback-Guided Domain Adaptation:** The damage estimation model developed from the training experiments may not directly generalize to the test and validation experiments because of differences in degradation characteristics among experiments. We therefore formulated the transfer from the training experiments to the unseen experiments as a domain

¹Yoshiki Kato et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

adaptation problem (Su & Lee, 2025). Although the ground-truth labels for the target experiments were not directly available, the competition leaderboard reported the MSE (Mean Squared Error) of submitted predictions. We used this scalar evaluation feedback as an indirect supervisory signal to iteratively refine the pseudo-labels of the target domain. The damage estimation model was then retrained using the updated pseudo-labels, allowing it to progressively adapt to the target domain degradation characteristics.

The following sections describe the proposed methods in detail and present the results of their application.

2. METHODOLOGY

2.1. Image analysis

This section describes the method used to detect damage from gear images and quantify damage progression. We calculated the area of the detected damage, represented by the number of pixels, and used it as the damage value of the corresponding tooth. Gear damage progression was represented by the maximum damage value among all teeth. The image analysis procedure is shown in Figure 1. The details of each step are as follows.

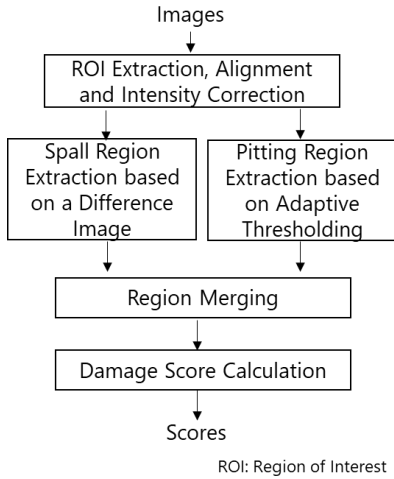


Figure 1. Flow of Image Analysis Procedure

- (1) **Cropping of the region of interest:** The locations where damage occurs are almost entirely limited to the central part of the image. Therefore, the region processed in the analysis was limited to the center of the image. Specifically, the region extending from 20% to 75% of the image height and from 20% to 80% of the image width was used as the region of interest. Image registration was used to correct positional displacement between runs, and brightness correction was also performed to reduce differences in brightness between runs.

- (2) **Detection of large-scale damage using image differencing:** Large-scale damage, or spalling, that occurs during the later stages of damage was detected by evaluating the difference from the preceding run. Regions in which the difference exceeded a predefined threshold were extracted. The magnitude of local changes in the image was then calculated using a Laplacian filter, and pixels whose mean Laplacian-filter value exceeded a predefined threshold were classified as spalling.
- (3) **Detection of small-scale damage using adaptive binarization:** Small-scale damage, or pitting, during the early stages of damage was extracted by detecting regions with lower brightness than their surroundings. Specifically, a Gaussian filter was applied to a 101×101 -pixel block centered on each target pixel. Pixels for which the difference between the filtered and original pixel values exceeded a predefined threshold were identified as pitting regions.
- (4) **Quantification:** The areas of the large-scale damage, or spalling, and the small-scale damage, or pitting, extracted above were combined and measured. The ratio of this area to the area of the processed image region, expressed as a percentage, was defined as the damage value for each tooth. The maximum damage value among all teeth was then defined as the damage value of the gear.

Examples of the extracted small- and large-scale damage regions are shown in Figure 2. The calculated damage value for each run is shown in Figure 3.

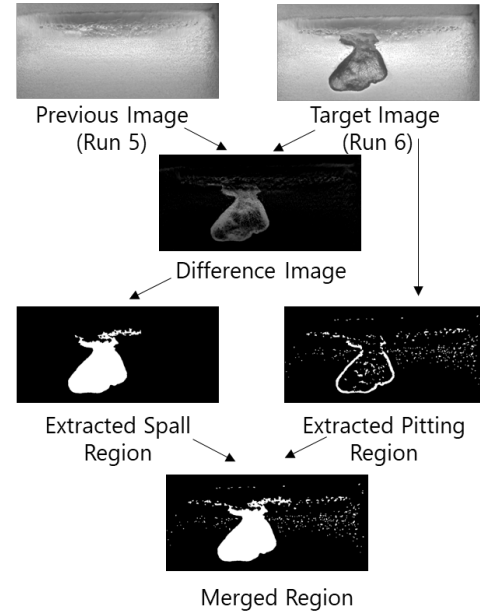


Figure 2. Examples of Extracted Damage Regions
(Exp-F, Tooth 18, Run 6)

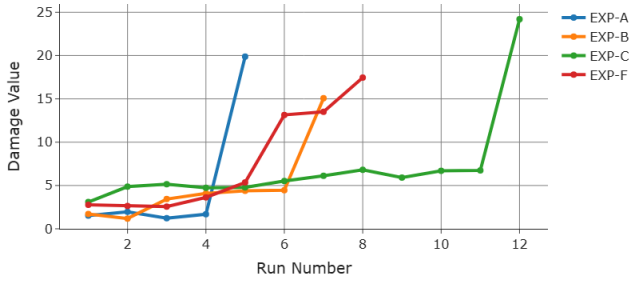


Figure 3. Calculated Damage Values

2.2. Condition Indicator Selection

Observation of the training images in the dataset showed that damage was not uniformly distributed across all 28 teeth. In some cases, pronounced spalling was observed on a small number of teeth, while most of the other teeth exhibited only relatively minor pitting. Therefore, we considered that a condition indicator sensitive not only to the average vibration change of the entire gear but also to local changes caused by a small number of severely damaged teeth would be effective for this task.

ALR is an indicator that uses changes in rotational harmonic components caused by tooth-to-tooth variation resulting from tooth-surface damage, such as pitting and spalling (Mark, Lee, Patrick & Coker, 2010). A previous study reported that ALR increases as pitting damage progresses (Mark, Wagner & Isaacson, 2023). Based on this characteristic, we adopted ALR as an indicator that could reflect the effects of locally progressed tooth-surface damage.

We also examined the TSA (Time Synchronous Averaging) waveform used to calculate ALR. Approximately 56 prominent peaks were observed in the TSA waveform over one revolution, corresponding to approximately twice the tooth count of 28 (Figure 4). This result is consistent with the TSA waveform having a periodic structure associated with tooth pitch.

$$x_{TSA}(k) = \frac{1}{M} \sum_{m=0}^{M-1} x(k + mN_r)$$

$$ALR = \frac{1}{N} \sum_{n=1}^N \left| \ln \left(\frac{|\alpha(n)|}{|\alpha_0(n)|} \right) \right|$$

Here, $x(\cdot)$ and $x_{TSA}(k)$ denote the original acceleration signal and the TSA signal, respectively. M is the number of rotations used for averaging, N_r is the number of samples per revolution, and k denotes the sample position within one revolution.

In addition, $\alpha(n)$ denotes the complex Fourier coefficient of the n th rotational harmonic of the TSA signal, and $\alpha_0(n)$

denotes the corresponding coefficient in the baseline condition. $|\alpha(n)|$ represents its amplitude, and N is the number of rotational harmonics used to calculate the ALR.

The vibration signals were resampled into 2,048 angular positions per revolution based on the encoder signal. When multiple acceleration samples corresponded to the same angular position within a revolution, their average was used. The resulting 2,048-point vectors were averaged over 100 consecutive revolutions to obtain each TSA signal. A baseline TSA waveform was constructed by taking the pointwise median of multiple TSA signals from the first run, and the baseline Fourier amplitudes used for the ALR calculation were obtained from its Fourier transform, as illustrated for one test in Figure 4.

Accordingly, the TSA signal for one revolution was divided into 28 equal angular sections corresponding to individual tooth pitches, and the local ALR was calculated for each section. In this study, this local ALR is referred to as tooth-pitch-wise ALR, whereas the conventional ALR is referred to as gear-level ALR.

As shown in Figure 5, when the gear-level ALR increased, particularly in Run 12, the tooth-pitch-wise ALR did not increase uniformly across all sections. Instead, pronounced increases were observed only in a subset of the tooth-pitch regions, as shown in Figure 6. This result suggests that the increase in gear-level ALR was associated with localized changes in the meshing response rather than with a uniform change over the entire gear circumference.

To confirm that the observed localized increase in ALR was not caused by the choice of tooth-pitch segmentation position, the segmentation boundary was shifted sample by sample over approximately one tooth pitch, and the tooth-pitch-wise ALR was recalculated for each offset. Figure 7 shows these results as a heatmap arranged according to circumferential position. The horizontal axis represents the circumferential position expressed in units of tooth pitch, while the vertical axis represents the progression of the experiment. Even when the segmentation position was shifted, increased ALR values were consistently observed around the same tooth-pitch region. This suggests that the localized increase in ALR observed in Figure 6 was not dependent on a particular offset, but was associated with a specific circumferential region of the gear.

On the other hand, we were unable to establish a relationship for directly quantifying the physical damage severity of each tooth from the tooth-pitch-wise ALR. We also could not identify a one-to-one correspondence between each tooth-pitch section and the actual physical tooth number. Therefore, tooth-pitch-wise ALR was used as an analysis tool to confirm the local damage sensitivity of ALR, while gear-level ALR, which provides a more stable time-series indicator, was used to calculate the final damage score.

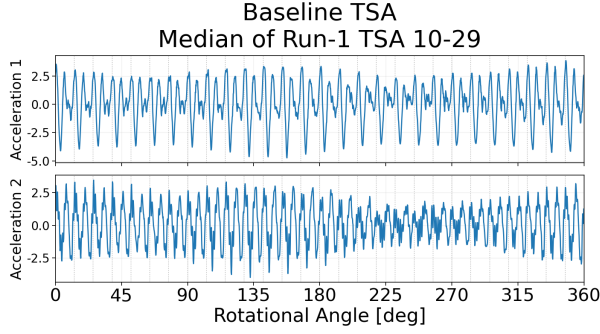


Figure 4. Angle-domain TSA waveform over one gear revolution (Exp-C).

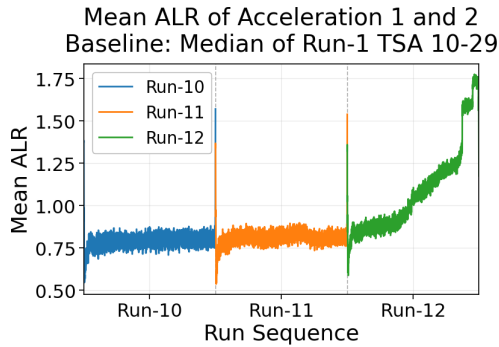


Figure 5. Gear-level ALR time history (Exp-C)

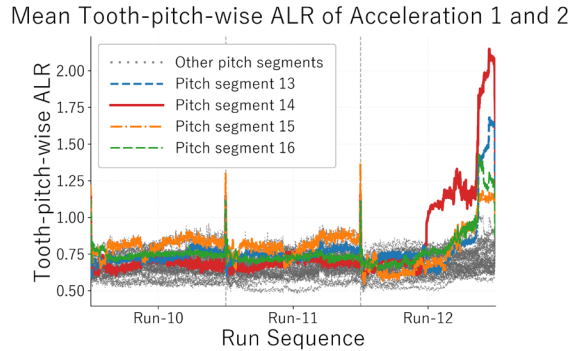


Figure 6. Tooth-pitch-wise ALR time histories (Exp-C)

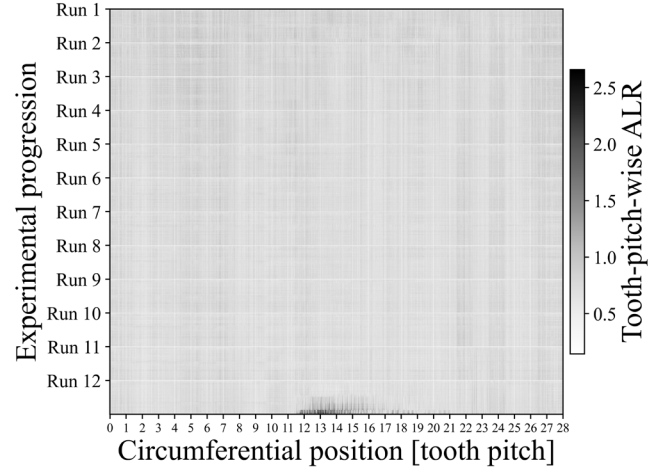


Figure 7. Evolution of Tooth-Pitch-Wise ALR over Experimental Runs (Exp-C).

2.3. Impulse Accumulation Model

As mentioned in Section 2.2, we selected ALR as the primary indicator because of its sensitivity to tooth-surface damage. However, for some runs with relatively minor damage, no clear change was observed in ALR.

Based on the observations, we introduced a complementary indicator based on the impulse accumulation model, for gradual changes in areas with relatively minor damage. The impulse accumulation model estimates damage value based on impulse accumulation value, where the impulse accumulation value is defined as the cumulative sum of the impulse magnitude.

During the early stages of gear damage, it is assumed that the magnitude of impulse increases as the damage progresses and that damage progresses more rapidly when the magnitude of impulse is greater. Therefore, we assume that, for each one-unit increase in the impulse accumulation value x , the resulting increment in damage value Δy is proportional to the current damage value y . Specifically,

$$\Delta y = ry$$

where r is a proportionality constant. If the initial value of the damage y is k , the relationship between the damage value and the impulse accumulation value x can be expressed as an exponential function:

$$y = k(1 + r)^x$$

The energy operator is a differential operator that estimates the instantaneous energy of a vibration signal (Li, Haiqi & Liwei, 2010).

We selected 'rms-e_op', or root mean square of the energy operator, included in the low-frequency dataset, as an indicator reflecting the magnitude of impulse and used its cumulative sum as the impulse accumulation value.

2.4. Ensemble model

As mentioned in Sections 2.2 and 2.3, we selected ALR as the primary indicator for severe damage while selecting impulse accumulation value as a complementary indicator for minor damage.

We trained a linear regression model to predict the damage value from the ALR indicator. We also trained an exponential regression model to predict the damage value from the accumulated impulse value. The damage value was estimated by combining the predictions of the two models using an ensemble approach.

In this competition, we used the following equation to calculate damage value:

$$Y = wY_{ALR} + (1 - w)Y_{IA}$$

$$w = X_{IA}/\text{Max}(X_{IA}),$$

where Y is the combined predicted damage value, and Y_{ALR} and Y_{IA} are the damage values predicted from the ALR and impulse accumulation models, respectively. w is the output of the weight function and X_{IA} is the impulse accumulation value.

The upper graph in Figure 8 shows the results of applying the two models described above to the training data. As shown in the figure, the predictions of the impulse accumulation model agree well with the ground truth during the early stages, whereas the predictions of the ALR model agree well with the ground truth during the later stages.

The lower graph in Figure 8 shows the result of applying the ensemble model, where the predictions agree well with the ground truth throughout the entire lifecycle.

Estimated Values on Training Data EXP-C

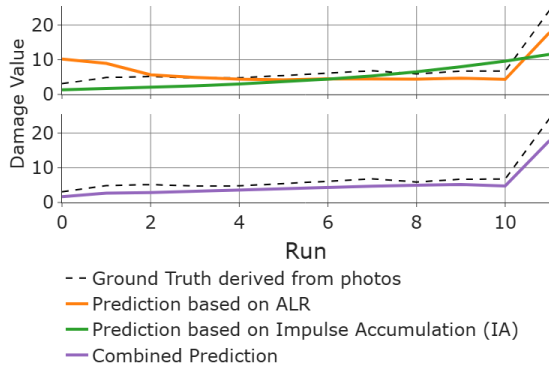


Figure 8. Results of Applying the Ensemble Model

2.5. Domain adaptation

This section describes the domain-adaptation method used to adapt the model trained on the training data to the test data.

In general, adapting a machine-learning model trained in one domain to another domain is referred to as domain adaptation. In this competition, we regarded the task of adapting a model trained using the training data to the test and validation data as a domain adaptation task and investigated an appropriate strategy. In domain adaptation tasks in which few or no ground-truth labels are available for the target domain, various methods characterized as **self-training domain adaptation** are often used (Lee, 2013; French, Mackiewicz & Fisher, 2018; Kumar, Ma & Liang, 2020). The strategy underlying in these methods is described below:

- 1) Apply a model trained in the source domain to data from the target domain and obtain labels. These labels are referred to as “pseudo-labels.”
- 2) Evaluate the pseudo-labels using some metric and update them in a direction that improves the evaluation result.
- 3) Retrain the model using the obtained pseudo-labels. Then, return to Step 1 and repeat the process.

Through these steps, a model trained in the source domain is gradually adapted to the target domain. In this competition, the leaderboard provided feedback in the form of MSE. We used MSE as the metric for evaluating the pseudo-labels.

As described above, we adopted the concept of self-training domain adaptation to develop our feedback-guided domain adaptation strategy. The processing flow is shown in Figure 9. The details of each process are as follows.

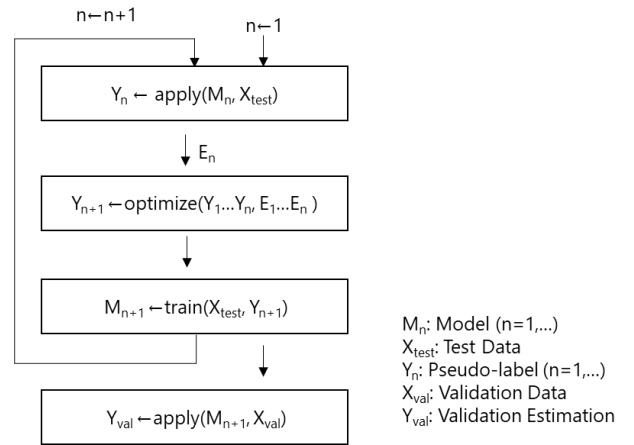


Figure 9. Flow of Domain Adaptation

- (1) **Generation and evaluation of pseudo-labels:** Let M be a model applied to test data X , producing pseudo-labels Y_1 . The pseudo-labels represent the damage values for

each run. When these labels are submitted to the leaderboard, an MSE value is returned.

- (2) **Optimization:** The pseudo-labels were optimized to reduce the MSE described above. Suppose that the results of n previous submissions are pseudo-labels $Y_1, \dots, Y_n$ and MSE values $E_1, \dots, E_n$ respectively. If the number of elements is d , then $d * E_i$ gives the squared Euclidean distance between pseudo-labels Y_i and the true labels Y_t .

In other words, the true labels Y_t exist at the intersection of the surfaces of hyperspheres whose centers are Y_i and whose radii are $\sqrt{d * E_i}$ for $i = 1, \dots, n$. Accordingly, we obtained the updated pseudo-labels Y_{n+1} by solving an optimization problem. The objective was to find a candidate label vector Y that minimizes the sum of the absolute differences between the Euclidean distance from Y to each hypersphere center Y_i and the corresponding radius $\sqrt{d * E_i}$, subject to the constraint that the elements of Y be monotonically increasing.

In general, when the number of trials n is smaller than the number of elements d , the solution is not unique, and the resulting pseudo-labels do not necessarily coincide with the true labels. Furthermore, the MSE of the updated pseudo-labels is not necessarily lower than that of the previous pseudo-labels. Nevertheless, in our experiments, repeated application of the procedure resulted in estimates that approached the true values in many cases.

- (3) **Retraining the model using the obtained labels:** The model was retrained using the updated labels.

By repeating these steps, the model was progressively adapted to the target domain.

3. RESULTS AND DISCUSSION

A model trained on training datasets B, C, and F¹ was adapted to the test data and subsequently applied to the validation data. The resulting MSE was 2.2, which was the best score among all participating teams.

To verify the effectiveness of domain adaptation, we conducted a simulation-based evaluation by assuming that the ground-truth labels for the test and validation data were available. When the model trained using the ground-truth labels established from the training data and image analysis was directly applied to the test and validation data, the respective MSE values were 6.83 and 20.73.

Next, we repeated the domain adaptation process described above six times. When the updated model was applied to the test and validation data, the MSE values were 0.65 and 2.13, respectively, confirming a substantial improvement over the model before adaptation (Figure 10, Table 1).

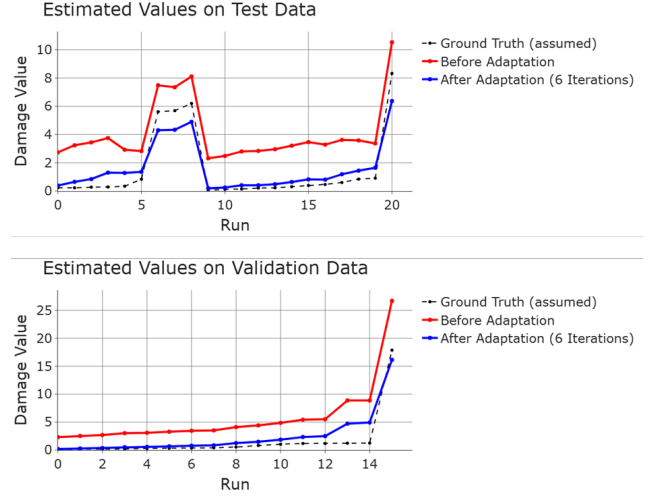


Figure 10. Results of Domain Adaptation

Table 1. Effects of Domain Adaptation

	Prediction Accuracy* (Mean Squared Error)		
	Before Adaptation	After 3 Iterations	After 6 Iterations
Test Data	6.83	4.28	0.65
Validation Data	20.73	10.23	2.13

*Prediction Accuracy was evaluated assuming a ground truth for test data and validation data. Therefore, these values do not necessarily match the leaderboard scores.

4. CONCLUSION

This paper presented a gear damage estimation method developed for the PHM 2026 Data Challenge. By combining an impulse accumulation model with the gear condition indicator ALR, we estimated the damage value over the entire lifecycle.

In addition, by adopting the concept of self-training domain adaptation and adapting the model trained using the training data to the test and validation data, we achieved highly accurate estimation in the competition.

¹ Dataset A was intentionally excluded from the training data because it exhibited substantially different characteristics from the other datasets in several variables, including torque.

REFERENCES

- Fan, Q., Zhou, Q., Wu, C., & Guo, M. (2017). Gear tooth surface damage diagnosis based on analyzing the vibration signal of an individual gear tooth. *Advances in Mechanical Engineering*, vol. 9, no. 6, pp. 1-14. doi:10.1177/1687814017704356
- French, G., Mackiewicz, M., & Fisher, M. (2018). Self-Ensembling for Visual Domain Adaptation. *ICLR*, 2018.
- Hines, J. A., & Mark, W. D. (2014). Bending-fatigue damage-detection on notched-tooth spiral-bevel gears using the average-log-ratio, ALR, algorithm. *Mechanical Systems and Signal Processing*, vol. 43, no. 1-2, pp. 44-56. doi:10.1016/j.ymssp.2013.09.013
- Kumar, A., Ma, T., & Liang, P. (2020). Understanding self-training for gradual domain adaptation. In *International conference on machine learning* (pp. 5468-5479). PMLR.
- Lee, D.-H. (2013). Pseudo-Label: The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks. *ICML Workshop: Challenges in Representation Learning*, 2013.
- Lee, J., & Su, H. (2025). Rethinking industrial artificial intelligence: A unified foundation framework. *International Journal of AI for Materials and Design*, 2(2), 56-68.
- Li, H., Haiqi, Z., & Liwei, T. (2010). Gear Fault Detection Based on Teager-Huang Transform. *International Journal of Rotating Machinery*. 2010. doi:10.1155/2010/502064.
- Luo, Y., Baddour, N., & Liang, M. (2019). Dynamical modeling and experimental validation for tooth pitting and spalling in spur gears. *Mechanical Systems and Signal Processing*, vol. 119, pp. 155-181. doi:10.1016/j.ymssp.2018.09.027
- Mark, W. D., Lee, H., Patrick, R., & Coker, J. D. (2010). A simple frequency-domain algorithm for early detection of damaged gear teeth. *Mechanical Systems and Signal Processing*, vol. 24, no. 8, pp. 2807-2823. doi:10.1016/j.ymssp.2010.04.004
- Mark, W. D., Wagner, M. E., & Isaacson, A. C. (2023). Detection and diagnosis of gear-tooth pitting damage by the Average-Log-Ratio ALR algorithm. *Mechanical Systems and Signal Processing*, vol. 186, Article 109911. doi:10.1016/j.ymssp.2022.109911.
- Sharma, V., & Parey, A. (2016). A Review of Gear Fault Diagnosis Using Various Condition Indicators. *Procedia Engineering*, 144, 253-263.
- Su, H., & Lee, J. (2024a). Machine learning approaches for diagnostics and prognostics of industrial systems using open source data from phm data challenges: A review. *International Journal of Prognostics and Health Management*, 15(2).
- Su, H., & Lee, J. (2024b). An advanced diagnostic model for gearbox degradation prediction under various operating conditions and degradation levels. *Annual Conference of the PHM Society*, vol. 16, no. 1.
- Su, H., & Lee, J. (2025). Transfer learning and ensemble learning for fault diagnosis using vibration signals. In *2025 IEEE International Conference on Prognostics and Health Management (ICPHM)* (pp. 1-9).