Wind Turbines Surface Damage Automatic Detection Using YOLOv8 with Specialized Backbone Modification

Aleksei Samarin, Anastasia Mamaeva,
Aleksei Toropov, Alina Dzestelova,
Egor Kotenko, Artem Nazarenko,
Elena Mikhailova, Valentin Malykh
ITMO University
St. Petersburg, Russia
avsamarin@itmo.ru, asmamaeva.science@gmail.com,
toropov.ag@hotmail.com, aldzestelova@gmail.com,
kotenkoed@gmail.com, aanazarenko@itmo.ru,
e.mikhailova@itmo.ru, valentin.malykh@phystech.edu

Alexander Savelev, Alexander Motyko
St. Petersburg Electrotechnical University “LETI”
St. Petersburg, Russia
algsavelev@gmail.com, aamotyko@etu.ru

***Abstract*—This work is devoted to incorporating specialized self-attention blocks into deep neural network-based models for detecting and quantifying damage across various components of wind turbines using images captured by unmanned aerial vehicle cameras. In our study, we introduce YOLOv8 backbone modification using a specialized self-attention mechanism, tailored to the specific characteristics of the input data. These modifications aim to enhance the model’s ability to accurately identify and assess the extent of damage from the complex visual data provided by drone imaging. To demonstrate the effectiveness of our proposed solution, we also publish an annotated dataset that we have compiled, which includes images of wind turbines captured by drone cameras. This dataset serves as a valuable resource for training and testing our models. Our solution, evaluated on test subsets of our dataset, has shown state-of-the-art results (mAP50-95 = 0.83234), surpassing most of widely used methods in performance metrics.**

I. Introduction

In the contemporary energy landscape, the depletion of conventional resources such as coal and oil looms large due to escalating demand. The exploitation of these fossil fuels inflicts increasingly adverse impacts on global ecology. Amidst a deepening climate crisis, the allure of renewable energy sources has magnified, with wind turbines emerging as a particularly promising alternative.

Globally, wind energy is plentiful, harnessing potential across diverse terrains including coastal, mountainous, and plain areas. Notably, wind energy is cost-effective and the operation of wind farms contributes zero greenhouse gas emissions, underscoring their environmental friendliness.

The 2023 Global Wind Report [1] by the Global Wind Energy Council, based in Brussels, Belgium, highlights that 77.6 gigawatts of new wind power capacity were commissioned worldwide in 2022. This expansion elevates the total installed capacity to 906 gigawatts, representing a 9% increase over the prior year. Regular inspections and routine maintenance are crucial to maintain wind farms’ safe and efficient operation, particularly in regions prone to severe weather conditions such as high winds, thunderstorms, and turbulence.

A critical aspect of sustaining high efficiency in power generation is the condition of wind turbine blades. Subject to constant exposure to natural elements like wind, sand, rain, snow, and seawater, these blades are vulnerable to wear and damage [2]. Proactive detection of surface damage on wind turbines is vital for continuous monitoring and ensures that the turbines are operational and safe ahead of any potential emergencies [3]. Damage not only compromises turbine performance but also has significant economic repercussions, including the costs of downtime for repairs or reduced efficiency due to minor damages. Given the size and complexity of wind turbine blades—typically, a 2 MW blade measures about 50 meters in length and weighs approximately 7000 kg — the financial impact of their repair is considerable [4].

Efforts are intensifying to reduce the operational and maintenance expenses associated with wind energy. Cutting-edge technologies such as automation [5], data analytics [6], [7], and artificial intelligence [8] are being integrated to streamline operational processes, enhance inspections, and refine maintenance routines, thereby reducing the reliance on human labor. These advancements are pivotal in optimizing the efficacy and sustainability of wind energy operations.

II. Problem Statement

Utilizing unmanned aerial vehicles, or drones, offers a transformative approach to evaluating wind turbines through cost-effective and routine inspections, leveraging high-resolution imagery. This method significantly reduces human involvement, thereby enhancing efficiency and reducing the costs of preventative maintenance [9]. Drones equipped with optical cameras can discern visible damage characteristics on wind turbine blades, such as leading-edge corrosion, surface cracks, and compromised lightning receptor mechanisms. However, despite the capabilities of drones and optical technology, internal damages often remain undetected [10], leading our research to focus primarily on surface damage.

The task of extracting and annotating damage information from a comprehensive set of detailed, high-resolution images requires considerable manual effort, which sustains high inspection costs. Manual analysis introduces potential for human error, yet regular drone monitoring can decrease the frequency of wind turbine maintenance, ultimately reducing overall energy expenses.

The challenge of detecting damage on wind turbines poses a quintessential machine learning problem, where the objective is to train a model capable of recognizing and classifying various types of damage. A wealth of research and benchmarks aims to refine detection methods, including the use of convolutional neural networks (CNNs) and other machine learning algorithms. This endeavor resides in the domain of computer vision, where models are trained using extensive datasets of damaged blade images, enabling them to autonomously detect and analyze damage in real-time. Recent advancements in deep learning have significantly boosted the performance and accuracy of these detection systems, allowing them to effectively handle diverse lighting conditions, perspectives, and damage extents.

This computer vision challenge involves training algorithms on extensive datasets featuring images of damaged blades, where the trained model can autonomously identify and assess damage in real time. The continual improvement in deep learning techniques enhances the efficacy and precision of these systems, equipping them to adeptly adapt to various lighting conditions, viewing angles, and damage sizes. The integration of machine learning in this area furthers the development of robust detection systems that can adapt to the dynamic operational conditions of wind turbines, reflecting a growing interest in innovative and efficient inspection and maintenance methods within the renewable energy sector.

Our study focuses on multiclass object detection, where the input data consists of three-dimensional tensors (images). The primary aim is to develop efficient methodologies that deliver catalogs of bounding boxes, each with corresponding confidence levels for the identified objects, improving the accuracy and reliability of the damage classification and localization processes on wind turbine blades, thereby enhancing the detection system's overall efficiency.

In our study, we analyzed various methods of object jetting regarding their effectiveness in solving the problem of detecting defects on the blades of wind power generators. We also upgraded the solution to outperform other baselines, allowing us to achieve the best results on our dataset, which we labeled and made publicly available.

Below we provide a detailed description of our work, starting with a description of baselines, a comparative analysis of their effectiveness, and a modification of the best approach to improve performance.

III. Related work

Modern machine learning and image processing technologies garner increasing attention in wind turbine blade damage detection and diagnosis.

A. Methods based on signal processing

The article ”Post-processing of ultrasonic signals for the analysis of defects in wind turbine blades using guided waves” [11] utilizes sophisticated ultrasonic signal processing techniques, such as discrete wavelet transform and variational mode decomposition, to scrutinize faults like breakages in wind turbine blades. The experimental results substantially enhance the accuracy of defect analysis. However, a significant constraint is the necessity to access the turbine blade from only one side, which may limit the thoroughness of defect detection across the entire blade surface.

In the study ”Damage detection in a laboratory wind turbine blade using techniques of ultrasonic Non-Destructive Testing (NDT) and Structural Health Monitoring (SHM)” [12], the effectiveness of ultrasonic NDT techniques and SHM systems is examined for identifying and localizing damages in a laboratory-scale wind turbine blade. This research considers two distinct methodologies: a nonlinear acoustic method and a guided wave monitoring method. Although the nonlinear acoustic method was found to be inadequately sensitive to detect experimentally induced damage, the guided wave monitoring method not only detected but also precisely localized damages using a ”novelty detector network” methodology. Initially, the directional wave tests did not yield successful results. Still, they ultimately provided invaluable insights into wave attenuation in structural elements, reinforcing the viability of deploying actuator and detector networks at appropriate densities for effective monitoring of wind turbine blades.

B. Methods based on machine learning algorithms

The study detailed in the cited paper [13] presents an innovative method for enhancing the parameters of support vector machines (SVM) using the harmonic search algorithm. This approach focuses on improving the accuracy of classifying and diagnosing vibration signals emanating from wind turbine blades. The research highlights the critical role of parameter optimization in augmenting classification accuracy. However, the method's complexity in signal analysis and the high associated costs might constrain its widespread practical deployment.

Conversely, the research described in the cited paper [14] explores the development of decision support tools essential for maintaining wind turbines. It utilizes decision tree learning algorithms to pinpoint faults within wind turbine structures, integrating telemetry data to detect anomalies effectively. This method is praised for its straightforward implementation and clarity, demonstrating a commendable efficiency with an error rate of less than 1%. Despite these advantages, it faces challenges such as the complexity of signal analysis and the high costs involved, which could hinder practical application. Additionally, the handling of large data volumes and their interpretation may present further complications, making the method's implementation in actual wind turbine conditions more challenging.

C. Methods based on computer vision and deep learning

The manuscript from the cited paper [15] presents a novel system that harnesses deep learning for the automated detection of damage on wind turbine blades. This system utilizes a faster R-CNN framework, which has been meticulously trained on a diverse dataset of images. The results indicate that this method achieves a level of accuracy comparable to that of manual expert evaluations in identifying both the location and type of damage on the blades. While this method marks a significant advancement, it is not without challenges, including a reliance on manual annotation of data and a necessity for sophisticated data augmentation techniques. These factors underscore the need for further refinements to fully operationalize the approach in practical settings.

In another groundbreaking study [16], researchers introduce a damage detection algorithm leveraging the YOLOv8 [17] architecture. Experimental validations of this algorithm demonstrate an impressive mean average precision (mAP) of 79.9%, significantly outperforming the Faster R-CNN [18] and SSD [19] models by 25.9% and 14%, respectively. The YOLOv8 model incorporates several advanced components such as the C2f-FocalNextBlock in the core network, ResNet-EMA in the neck network, and a slim-neck structure, which are engineered to enhance feature extraction, foster inter-spatial interactions, and facilitate the integration of features across various scales. Despite the algorithm's superior performance, its practical deployment may encounter hurdles due to the management of numerous parameters and the substantial demand for computational resources.

IV. Proposed solution

A. Basic YOLOv8 architecture

This section outlines the structure of the YOLOv8 architecture (baseline with the best performance on our dataset, according to Sec. V), providing a foundation for algorithm design and the modifications that we applied to it to increase the performance on our dataset. The YOLOv8 architecture mainly consists of four components: input, backbone, neck, and head. The structure of the YOLOv8 algorithm is illustrated in Fig. 1.

Input

In the study involving the YOLOv8 algorithm, the preprocessing of input images is a critical step due to the variance in image dimensions within the dataset. This preprocessing involves scaling adjustments to enhance the efficiency of image processing, standardizing the dimensions of input images to a uniform size of 640 × 640 pixels. This standardization minimizes the necessity for adding black borders, thereby optimizing the image presentation for subsequent processing stages.

Moreover, the YOLOv8 framework incorporates advanced features such as adaptive anchor frame calculation and mosaic data enhancement. These functionalities allow users to tailor data enhancement techniques to best fit their specific dataset needs. Particularly, the mosaic data enhancement technique enriches the dataset by randomly scaling and cropping four images to create a single composite image, thereby significantly bolstering the model's robustness against varied data inputs.

During the training phase of the network, YOLOv8 generates predicted bounding boxes from the initial anchor boxes. These predictions are then aligned with the ground truth boxes to compute the loss function. Through iterative backpropagation, the network parameters are continuously refined, aiming to minimize this loss, which in turn enhances the accuracy and reliability of the model in real-world applications.

Backbone

The YOLOv8 algorithm's backbone network plays a pivotal role in extracting fundamental features from the target objects. This network is structured around three primary modules: Conv, C2f, and SPPF. The Conv module is composed of three integral components: Conv2d, Batch Normalization (BN) [21], and the SiLU [22] activation function. It efficiently achieves the desired padding effect through the use of autopad(k, p), optimizing the handling of image boundaries.

The design of the C2f module draws inspiration from the C3 and ELAN modules. This module enhances the YOLOv8's capability to capture richer gradient flow information, which is crucial for effective feature learning while maintaining a streamlined, lightweight architectural design.

Lastly, the SPPF module, informed by the principles of SPP-Net [23], undergoes a strategic redesign. Instead of utilizing a single large-sized pooling core as in the traditional SPP module, it employs a series of smaller-sized pooling cores arranged in a cascade. This innovative configuration preserves the original functionality of spatial pyramid pooling while optimizing the module for better performance and efficiency in feature processing.

Neck

The YOLOv8 algorithm's neck network architecture ingeniously integrates the principles of the Feature Pyramid Network (FPN) [24] and the Path Aggregation Network (PAN) [25], forming a composite FPN+PAN structure. This neck network acts as a critical conduit between the backbone and the prediction networks, effectively amalgamating features from different layers to enhance target detection capabilities.

FPN is adept at synthesizing high-level features, which possess low resolution but carry high semantic value, with low-level features, which are high resolution with low semantic value. This combination fosters the generation of semantically enriched features across all scales, optimizing the network's ability to interpret complex images.

Conversely, PAN enhances the network by implementing adaptive feature pooling, which accelerates the information flow from the lower to the upper layers, reinforcing the signal integrity at the foundational layer and improving the overall feature hierarchy. PAN further refines this process by ensuring that highly localized features are propagated from the bottom

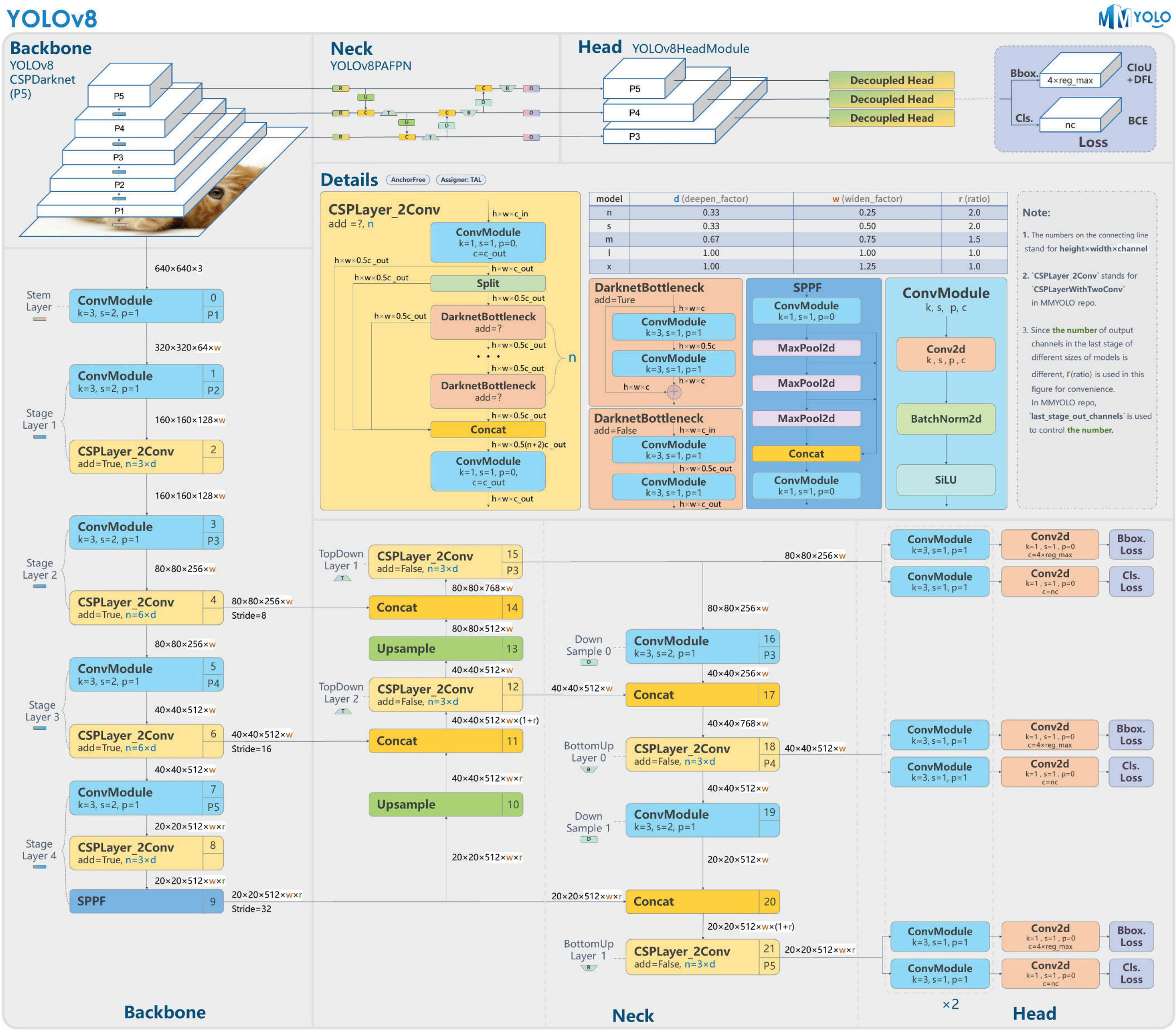


Fig. 1. YOLOv8-P5 model structure [20]

upwards, which reduces the distance information needs to travel between the lowest and highest layers of the network, thereby streamlining the feature integration process.

Additionally, within the YOLOv8's neck network, the traditional C3 module is replaced with the more advanced C2f module, and it discards the convolutional structure previously used in the PAN-FPN upsampling phase seen in YOLOv5. This revision not only simplifies the network architecture but also enhances its efficiency in handling feature pooling and propagation, ensuring that each layer contributes effectively to the final prediction accuracy.

Head

The head architecture of the YOLOv8 algorithm features a decoupled head structure, which represents an advanced approach to separating the tasks of classification and detection. This segmentation utilizes the concept of Distributional Focal Loss (DFL) to enhance specificity and accuracy in handling different tasks. The algorithm's head comprises three distinct detection layers, each tailored to manage different types of anchors, which vary in aspect ratios and are derived directly from the neck network. These layers are specifically designed for precise prediction and regression of target objects.

Moreover, the adaptability of the YOLOv8 algorithm is evident in its anchor boxes, which are dynamically responsive to the dataset being used. These anchor boxes can automatically adjust their dimensions based on the characteristics of various datasets, ensuring that the detection process is both flexible and robust, tailored to meet the specific needs of diverse input data. This adaptability plays a crucial role in optimizing the algorithm's performance across different scenarios, making it highly effective in real-world applications.

B. Backbone modification

To improve performance when solving the specific problem, we used (a backbone modification approach similar to that presented in article [26]. To implement this, we also changed the backbone of the model, using an encoder with a specially modified type of attention for the most stable focus on image areas with the most likely localization of defects (a similar idea used in [27]).

As a baseline for our backbone, we used several types of DNNs (such as VGG [28], Inception [29], ResNet [30], MobileNet [31], EfficientNet [32], CoAtNet [33], etc.) We also used attention blocks as an additional constructive element for listed baseline encoders to increase the performance of our solution. It should be noted that attention block insertion allows to boost up the overall DNNs performance on a wide range of specific problems [34]–[38].

However, several works have shown how the modification of the basic attention mechanism in specific cases gives an increase in the values of quality metrics [38].

Thus, the general view of our modified backbone is shown in Fig. 2.

Moreover, further we will take a closer look at the modifications of the attention block that we used in this work for the domain under consideration.

C. Customized attention block

1) Basic non-local block (Basic NLB) construction: First, let's introduce general notation for specifying the structure of non-local blocks using the example of a basic attention block. Here we will use the notation used in the work [38].

The functional schema of this attention mechanism, illustrated in Fig. 3, is well acknowledged.

Initially, the input tensor is segmented into rectangular grids through grid-based tokenization. Following this, each grid element (token) is converted into a tensor of equivalent form x_i, where $|x_i| = N = n * n$, and $dim(x_i) = d$. Subsequently, three sets of tensors Q, K, V are derived through trainable projection operations $(f(x), g(x), h(x))$ applied to the original tensor set x_i, resulting in

$$Q = f(x_i), K = g(x_i), V = h(x_i).$$

Following this, similarity coefficients between each element $q_i \in Q$ and other tensors are computed by evaluating dot products $q_i * k_j \ \forall j \in [1..n]$. Normalization, such as $SoftMax$ normalization, is then applied to these dot products, often preceded by division by the square root of the dimension $dim(k_i) = d_k$, yielding

$$\alpha_i = SoftMax(q_i k_1/d_k, q_i k_2/d_k, ..., q_i * k_n/d_k).$$

Subsequently, an attention map is constructed by linearly combining these coefficients α_i with the respective tensors v_i and applying a trainable projection operation $v(x)$ to the resultant expression:

$$o_i = v(\sum_{j \in [1..n]} (\alpha_i * v_j)).$$

Alternatively, this sequence of operations can be represented in matrix form, where Q, K, V are denoted in matrix format as

$$Q = [f(x_1), ..., f(x_n)],$$
$$K = [g(x_1), ..., g(x_n)],$$
$$V = [h(x_1), ..., h(x_n)].$$

Consequently, the resultant attention map can be computed as

$$O = v(SoftMax(\frac{QK^T}{\sqrt{d_k}})).$$

The resulting attention map is employed to reweight the elements of the input tensor $X = [x_1, ..., x_n]$ in conjunction with the concept of the residual connection, yielding: $Y = WO + X$.

2) Non-local block customizations: Since our task is related to the localization of objects (defects) in images of a specific visual series, we also upgraded the visual feature encoders in such a way as to highlight areas as efficiently as possible, taking into account the general visual characteristics of the background. To implement this idea, we modified the basic structure of the attention block as part of the backbone of the model in the ways presented below.

Homogeneous background heuristic

The first heuristic was based on the observation of color uniformity for parts of wind generators. The fact is that the defects we are looking for are located on the blades of wind generators, the texture and color range of which are very uniform. Thus, it is convenient to modify the encoder in such a way as to highlight inhomogeneities in homogeneous areas of images. That is why we modify the self-attention map according to the following post-processing:

$$O^* = O + w_{O1} * (O - w_{O2} * \frac{O}{dim(O)}),$$

where O is attention map from basic non-local block description, w_{O1} and w_{O1} are trainable parameters, O^* is the modified attention map that is evaluated after O and is used instead O in the self-attention module. So the result modified non-local block scheme is presented in Fig. 4.

Similarity-based homogeneous background heuristic

To implement the second heuristic, we added another element-wise post-processing to the components of the attention map, to further take into account the heterogeneity of the texture of the cavities of wind generators.

We add another posterior component to the attention map elements evaluation expression that reduces the value of attention map components whose relational original image token value is similar to other grid tokens of the original image:

$$o_i^{**} = v(\sum_{j \in [1..n]} (\alpha_i * v_j)) - w_{O3} * \frac{\sum_{j \in [1..n], j \neq i} \frac{1}{dist(x_i, x_j)}}{n-1},$$

where α_i, v_i, o_i, n, j , i denotes the same entities as above, w_{O3} denotes a trainable parameter, $dist$ denotes $L_1 + L_2$ distance in a vector space. So the resulting attention block modification is presented in Fig. 5.

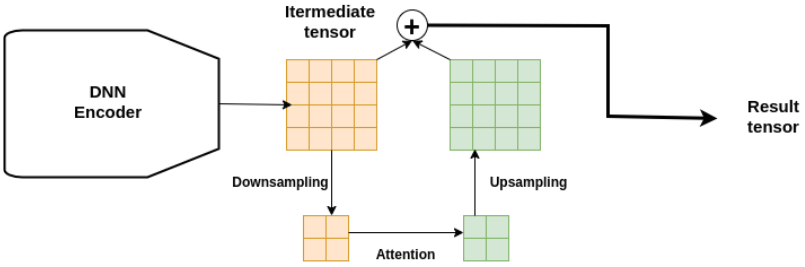


Fig. 2. Modified backbone structure.

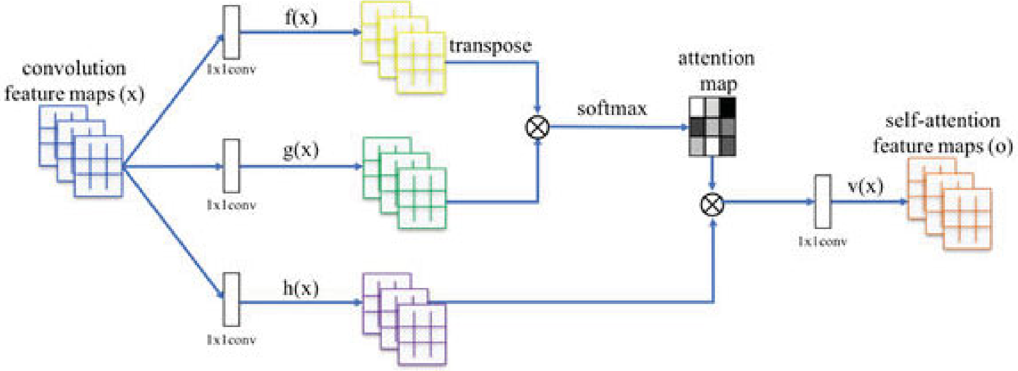


Fig. 3. Original non-local block scheme [38].

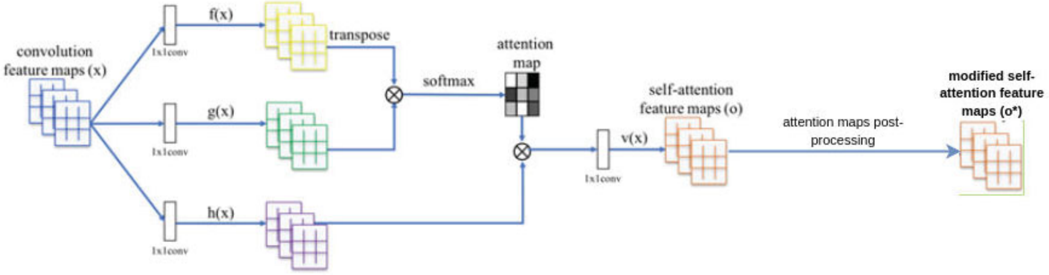

convolution feature maps (x)
f(x)
g(x)
h(x)
1x1conv
1x1conv
1x1conv
transpose
softmax
attention map
v(x)
1x1conv
self-attention feature maps (o)
attention maps post-processing
modified self-attention feature maps (o*)

Fig. 4. Modified non-local block scheme with homogeneous background heuristic.

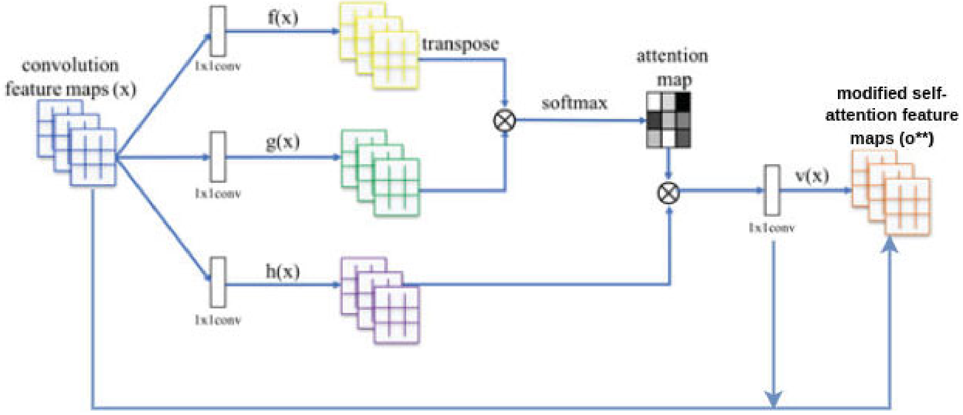


Fig. 5. Modified non-local block scheme with similarity-based homogeneous background heuristic.

V. Experimental setup

To test the effectiveness of the described architecture modifications and analogs, we conducted a series of experiments on a specially created and labeled dataset, which we made publicly available. Below is a description of the data set and experiment base.

A. Dataset description

In the context of this study, we utilized a publicly accessible dataset that comprises images derived from drone inspections conducted on the Nordtank turbine during the years 2017 and 2018. This dataset is made available in the Mendeley public repository [39] and includes a total of 561 high-resolution images captured from various angles and distances, offering an exhaustive visual assessment of the turbine's surface.

The dataset encompasses images with an average size of 15.68 megapixels, while the resolutions of these images vary between 1.23 and 15.68 megapixels. The images predominantly feature a median aspect ratio of 5280 × 2970, aligning with the widescreen format, which is particularly conducive for detailed visual analysis.

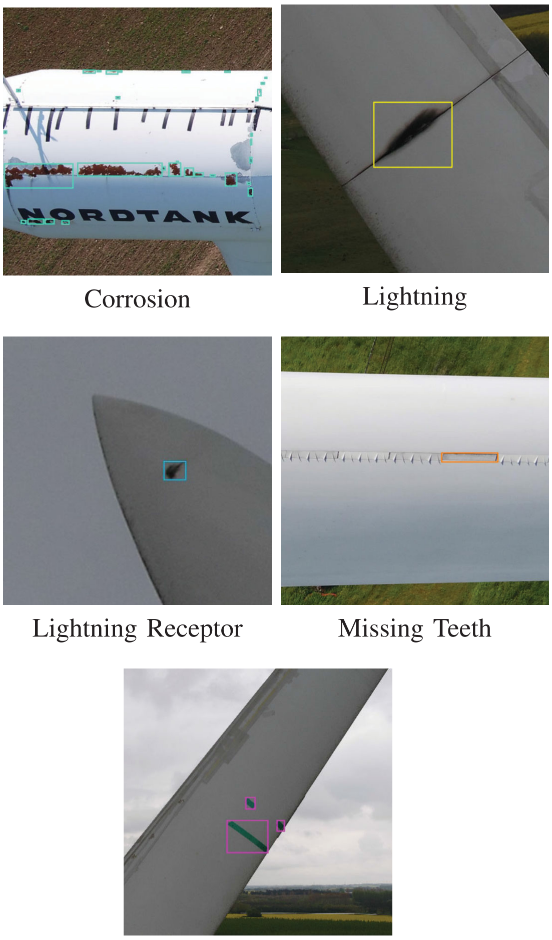
During the meticulous annotation process, several types of damage to the turbine were systematically identified and cataloged, enhancing the dataset's utility for in-depth research and analysis. These types of damage are presented in Fig. 6.

- **Corrosion**: Areas susceptible to corrosion include sections where oxidation of metal surfaces has occurred, which can lead to a reduction in the structural integrity of the turbine.
- **Lightning**: Damage caused by lightning strikes includes electrical discharge marks and thermal damage to the turbine surface.
- **Lightning Receptor**: Special devices designed to receive lightning strikes also need to be monitored for damage to ensure they function effectively.
- **Missing Teeth**: Defects associated with missing teeth on gears and other mechanical parts of a turbine can lead to mechanical failures.
- **Patches**: Areas where patches have been installed to temporarily repair damage should also be tracked to assess their condition and the need for further repair.

Annotation of the dataset was meticulously performed manually utilizing the Roboflow [40] platform. The datasets were methodically partitioned into training, testing, and validation subsets, with allocations of 70% of the images designated for training, 20% set aside for testing, and the remaining 10% dedicated to validation. This structured distribution ensures a comprehensive and rigorous evaluation of the model's performance across different stages of development.

B. Experimental environment

This study employed the PyTorch deep learning framework to conduct experimental analyses. The training and validation phases of the research were executed on an NVIDIA Geforce RTX 4070 GPU, equipped with 12GB of memory, providing robust computational power and efficiency necessary for handling complex neural network tasks. However, the neural network solutions proposed in this article can be launched using graphics accelerators of the same class and the amount of GPU RAM.



Corrosion

Lightning

Lightning Receptor

Missing Teeth

Patches

Fig. 6. The examples of manually-annotated damage to wind turbines.

C. Experimental results

The findings are presented in Table I for the detection task. The YOLOv8 modification with ResNet-101 backbone with similarity-based homogeneous background heuristic-based non-local block modification (similarity-based homogeneous background heuristic) outperformed counterparts on the test subsample of the dataset.

VI. Conclusion

In this paper, we presented modifications to self-attention blocks as integral components of a deep neural network designed to address the challenge of detecting damage to various components of wind turbines using images captured by unmanned aerial vehicle (UAV) cameras. To evaluate the effectiveness of our approach, we compiled, annotated, and publicly released a specialized dataset of wind turbine images wind turbines captured by drone cameras. The modifications we implemented enhanced the performance of the underlying

TABLE I. TOP EXPERIMENTAL RESULTS

Model	Precision	Recall	mAP50	mAP50-95
SSD-300	0.71452	0.47231	0.62784	0.38954
SSD-512	0.72345	0.45123	0.60897	0.37896
Faster R-CNN + MobileNetV3	0.73129	0.48231	0.63984	0.39743
Faster R-CNN + ResNet-50	0.74321	0.47234	0.63812	0.39567
Cascade R-CNN [41] + ResNet50	0.75432	0.48123	0.65498	0.40234
Cascade R-CNN + ResNet101	0.76123	0.49123	0.66897	0.41467
RetinaNet [42] + ResNet-50	0.76549	0.49531	0.66184	0.40965
RetinaNet + ResNet-101	0.78123	0.51234	0.69123	0.44789
EfficientDet-D1 [43]	0.79345	0.52345	0.70789	0.45412
EfficientDet-D3	0.80567	0.53412	0.71234	0.45967
YOLOv8n	0.8179	0.52713	0.56018	0.32756
YOLOv8s	0.83453	0.54810	0.61048	0.36240
YOLOv8m	0.81665	0.50825	0.59219	0.34496
Ours, YOLOv8 + CoAtNet	0.85432	0.57891	0.78345	0.53287
Ours, YOLOv8 + ResNet-101(NLB)	0.87456	0.61934	0.82945	0.59876
Ours, YOLOv8 + ResNet-101(NLB-Hom)	0.87943	0.62145	0.82897	0.59712
Ours, YOLOv8 + ResNet-101(NLB-Sim)	**0.88567**	**0.62178**	**0.83234**	**0.60789**

deep neural network, as evidenced by the improved values of target quality metrics on our published dataset.

Our work demonstrates the potential of tailored self-attention mechanisms to refine feature extraction processes, enabling more precise detection and characterization of anomalies in complex structures like wind turbines. The approach leverages the unique capabilities of deep learning to interpret intricate patterns and irregularities that may signify damage, thereby offering a robust tool for maintenance and safety assessments in renewable energy installations.

Furthermore, the techniques and findings from this study can be extended beyond the scope of wind turbines to other critical infrastructure components, such as bridges, buildings, and other mechanical hard-to-reach systems where safety and integrity are paramount. By applying similar modifications to self-attention blocks within neural networks, it is possible to enhance the detection and diagnosis of structural deficiencies or damages in a wide range of applications, ultimately contributing to more effective predictive maintenance strategies and ensuring the longevity and reliability of key infrastructural elements. Another layer of further work can be focused on working with combined datasets to universalize the method and additional discriminative damage analysis.

REFERENCES

[1] "Global wind report 2023 — global wind energy council." [Online]. Available: https://gwec.net/globalwindreport2023/

[2] T. Regan, C. Beale, and M. Inalpolat, "Wind turbine blade damage detection using supervised machine learning algorithms," *Journal of Vibration and Acoustics*, vol. 139, 06 2017.

[3] G. Skrimpas, K. Kleani, N. Mijatovic, C. Sweeney, B. Jensen, and J. Holboell, "Detection of icing on wind turbine blades by means of vibration and power curve analysis," *Wind Energy*, vol. 19, pp. n/a–n/a, 12 2015.

[4] D. Katsaprakakis, N. Papadakis, and I. Ntintakis, "A comprehensive analysis of wind turbine blade damage," *Energies*, vol. 14, p. 5974, 09 2021.

[5] C. Stock-Williams and S. Krishnaswamy, "Automated daily maintenance planning for offshore wind farms," *Renewable Energy*, 09 2018.

[6] K. Zhou and S. Yang, "Big data driven smart energy management: From big data to big insights," *Renewable and Sustainable Energy Reviews*, vol. 56, pp. 215–225, 04 2016.

[7] M. Cañizo, E. Onieva, A. Conde, S. Charramendieta, and S. Trujillo, "Real-time predictive maintenance for wind turbines using big data frameworks," 06 2017, pp. 70–77.

[8] S. Jha, J. Bilalovic, A. Jha, N. Patel, and H. Zhang, "Renewable energy: Present research and future scope of artificial intelligence," *Renewable and Sustainable Energy Reviews*, vol. 77, pp. 297–317, 09 2017.

[9] G. Morgenthal and N. Hallermann, "Quality assessment of unmanned aerial vehicle (uav) based visual inspection of structures," *Advances in Structural Engineering*, vol. 17, pp. 289–302, 03 2014.

[10] X. Chen, "Fracture of wind turbine blades in operation—part i: A comprehensive forensic investigation. wind energy," 6 2018.

[11] K. Tiwari and R. Raišutis, "Post-processing of ultrasonic signals for the analysis of defects in wind turbine blade using guided waves," *The Journal of Strain Analysis for Engineering Design*, vol. 53, 05 2018.

[12] K. Yang, J. Rongong, and K. Worden, "Damage detection in a laboratory wind turbine blade using techniques of ultrasonic ndt and shm," *Strain*, vol. 54, p. e12290, 08 2018.

[13] H.-C. Sun and Y.-C. Huang, "Support vector machine for vibration fault classification of steam turbine-generator sets," *Procedia Engineering*, vol. 24, p. 38–42, 12 2011.

[14] I. Abdallah, V. Dertimanis, C. Mylonas, K. Tatsis, E. Chatzi, N. Dervilis, K. Worden, and A. Maguire, "Fault diagnosis of wind turbine structures using decision tree learning algorithms with big data," 06 2018.

[15] A. Shihavuddin, X. Chen, V. Fedorov, A. Christensen, N. Riis, K. Branner, A. Dahl, and R. Paulsen, "Wind turbine surface damage detection by deep learning aided drone inspection analysis," *Energies*, vol. 12, 02 2019.

[16] L. Liu, P. Li, D. Wang, and S. Zhu, "A wind turbine damage detection algorithm designed based on yolov8," *Applied Soft Computing*, vol. 154, p. 111364, 02 2024.

[17] M. Sohan, T. Ram, and V. Ch, *A Review on YOLOv8 and Its Advancements*, 01 2024, pp. 529–545.

[18] R. Gavrilescu, C. Zet, C. Foșalău, M. Skoczylas, and D. Cotovanu, "Faster r-cnn:an approach to real-time object detection," in *2018 International Conference and Exposition on Electrical And Power Engineering (EPE)*, 2018, pp. 0165–0168.

[19] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. Berg, "Ssd: Single shot multibox detector," vol. 9905, 10 2016, pp. 21–37.

[20] "Openmmlab. yolov8 configuration." [Online]. Available: https://github.com/open-mmlab/mmyolo/tree/main/configs/yolov8

[21] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ser. ICML'15. JMLR.org, 2015, p. 448–456.

[22] S. Elfwing, E. Uchibe, and K. Doya, "Sigmoid-weighted linear units for neural network function approximation in reinforcement learning," *Neural Networks*, vol. 107, 01 2018.

[23] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *Computer Vision –*

ECCV 2014, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 346–361.

[24] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 936–944.

[25] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8759–8768.

[26] M. Zhang, Z. Wang, W. Song, D. Zhao, and H. Zhao, "Efficient small-object detection in underwater images using the enhanced yolov8 network," *Applied Sciences*, vol. 14, no. 3, 2024. [Online]. Available: https://www.mdpi.com/2076-3417/14/3/1095

[27] A. Samarin, A. Toropov, A. Dzestelova, A. A. Nazarenko, E. Kotenko, E. Mikhailova, A. Savelev, and A. Motyko, "Specialized non-local blocks for recognizing tumors on computed tomography snapshots of human lungs," in *35th Conference of Open Innovations Association, FRUCT 2024, Tampere, Finland, April 24-26, 2024*. IEEE, 2024, pp. 659–664. [Online]. Available: https://ieeexplore.ieee.org/document/10516343

[28] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[29] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015, pp. 1–9.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 770–778.

[31] 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," 04 2017.

[32] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," 05 2019.

[33] Z. Dai, H. Liu, Q. Le, and M. Tan, "Coatnet: Marrying convolution and attention for all data sizes," *CoRR*, 2021.

[34] A. Samarin, A. Savelev, and V. Malykh, "Two-staged self-attention based neural model for lung cancer recognition," in *2020 Science and Artificial Intelligence conference (S.A.I.ence)*, 2020, pp. 50–53.

[35] A. Samarin, A. Savelev, A. Toropov, A. Dzestelova, V. Malykh, E. Mikhailova, and A. Motyko, "One-staged attention-based neoplasms recognition method for single-channel monochrome computer tomography snapshots," *Pattern Recognition and Image Analysis*, vol. 32, pp. 645–650, 10 2022.

[36] A. Samarin, A. Savelev, A. Toropov, A. Dzestelova, V. Malykh, and E. Mikhailova, *Prior Segmentation and Attention Based Approach to Neoplasms Recognition by Single-Channel Monochrome Computer Tomography Snapshots*, 08 2023, pp. 561–570.

[37] A. Samarin, A. Savelev, A. Toropov, A. Dzestelova, V. Malykh, E. Mikhailova, and A. Motyko, "One-staged attention-based neoplasms recognition method for single-channel monochrome computer tomography snapshots," *Pattern Recognition and Image Analysis*, vol. 32, pp. 645–650, 10 2022.

[38] A. Samarin, A. Toropov, A. Dzestelova, A. Nazarenko, E. Kotenko, E. Mikhailova, A. Savelev, and A. Motyko, "Specialized non-local blocks for recognizing tumors on computed tomography snapshots of human lungs," in *2024 35th Conference of Open Innovations Association (FRUCT)*, 2024, pp. 659–664.

[39] C. X. Shihavuddin, A., "Dtu—drone inspection images of wind turbine," 2017.

[40] "Roboflow platform." [Online]. Available: https://roboflow.com/

[41] Z. Cai and N. Vasconcelos, "Cascade r-cnn: Delving into high quality object detection," 06 2018, pp. 6154–6162.

[42] R. Del Prete, M. D. Graziano, and A. Renga, "Retinanet: A deep learning architecture to achieve a robust wake detector in sar images," in *2021 IEEE 6th International Forum on Research and Technology for Society and Industry (RTSI)*, 2021, pp. 171–176.

[43] M. Tan, R. Pang, and Q. Le, "Efficientdet: Scalable and efficient object detection," 06 2020, pp. 10 778–10 787.