JOURNAL
COMPUTER SCIENCES INSTITUTE

JCSI 37 (2025) 470–475
Received: 30 June 2025
Accepted: 7 September 2025

The use of machine learning to classify symbols on cultural monuments to identify their origin and historical period

Zastosowanie uczenia maszynowego do klasyfikacji symboli na zabytkach kultury w celu identyfikacji ich pochodzenia i epoki historycznej

Igor Pajura*, Sylwester Korga

Department of Computer Science, Lublin University of Technology, Nadbystrzycka 36B, 20-618 Lublin, Poland

Abstract
Identifying and cataloguing cultural heritage objects is a time consuming process that requires expert knowledge. This study explores the application of deep learning models, specifically YOLOv8 and ResNet50, to classify historic buildings by historical epoch and country of origin, respectively. This research was conducted using a dataset of 3,200 images, which featured monuments categorized into four separate historical periods and representing four distinct nations. YOLOv8 detected buildings and classified them into historical epochs while ResNet50 was used for classifying the country of origin. The analysis demonstrated that models achieved a notable degree of effectiveness in identifying both the architectural epochs and the countries of origin.

Keywords: Deep Learning; YOLOv8; ResNet50

Streszczenie
Identyfikacja i katalogowanie obiektów dziedzictwa kulturowego to czasochłonny proces wymagający specjalistycznej wiedzy. Niniejsze badanie dotyczy zastosowania modeli głębokiego uczenia, a konkretnie YOLOv8 i ResNet50, do klasyfikacji zabytkowych budynków według epoki historycznej i kraju pochodzenia. Badania przeprowadzono na zbiorze danych zawierającym około 3200 obrazów przedstawiających zabytki podzielone na cztery odrębne okresy historyczne i reprezentujące cztery różne kraje. Model YOLOv8 wykrywał budynki i klasyfikował je według epok historycznych, natomiast model ResNet50 służył do klasyfikacji kraju pochodzenia. Analiza wykazała, że modele osiągnęły znaczący stopień skuteczności w identyfikacji zarówno epok architektonicznych, jak i krajów pochodzenia.

Słowa kluczowe: Głębokie uczenie; YOLOv8; ResNet50

Email address: s95514@pollub.edu.pl

1. Introduction

The process of identifying and cataloging cultural heritage objects in most cases requires specialized knowledge and often the analysis itself is time-consuming. These processes are essential and enable preservation and a better understanding of humanity's past. The advancing digitization of cultural monuments leads to an increase in the amount of digital data, which in turn creates a need for efficient methods that allow automation of the classification. In recent years machine learning techniques like deep neural networks have seen significant development. They were an important step in the development of computer vision field. These models have achieved high efficiency in the tasks of object recognition and classification while outperforming previous algorithms. The application of machine learning in the analysis of cultural heritage opens new possibilities for automating cataloging processes and identifying their origin. This article aims to see if the application of machine learning can be applied to the classification of symbols on cultural monuments to identify their origin and historical era. A breakthrough in image recognition came with representation learning, which, as noted in [1], allows machines to automatically discover features from raw data, unlike older models that required manually designed filters. Traditional machine learning methods often required converting images into one-dimensional vectors, which, as indicated in comparative analyses [2], led to the loss of spatial information. Modern methods, explored in [1] among others, learn hierarchical representations, where higher layers amplify relevant aspects of the input data [1]. Even in deep approaches, as noted in texture classification studies [3], feature selection remains crucial, although Convolutional Neural Networks (CNNs) can learn them automatically. CNNs have become the foundation of deep learning in computer vision. Their architecture, as described in reviews such as [4], consists of convolutional layers performing convolution operations using learning filters [4] and pooling layers reducing the dimensions of feature maps [4]. Deep CNNs effectively capture features at various levels of abstraction [3], integrating them in a comprehensive manner from input to output, which is a characteristic feature of multilayer architectures, as highlighted in [5] and [6]. In object detection, the development of CNNs has led to the emergence of effective architectures, divided into single-stage and two-stage methods, which has been identified as a fundamental division in works such as [7]. Two-stage methods, exemplified by region-based Convolutional Neural Networks (R-CNN) [8], first generate regions of interest

(RoI) – which, as indicated by research [9], is the first stage of these detectors – and then make predictions. In contrast, single-stage detectors, popularized by YOLO, formulate the problem as regression to bounding boxes and class probabilities [10], analyzing all spatial proposals at once, as summarized in [9]. In image classification, VGG architectures, as discussed with respect to cultural heritage in [11], utilize small filter sizes for efficient information extraction. Subsequently, the use of 1x1 convolutions to impose dimensionality reduction prior to more expensive operations, as introduced in [8], makes it possible to realize deeper and broader networks [8]. The development encompasses attention mechanisms, a good example of this being the Residual Attention Network that creates features for the important regions of the image using attention modules. The deep neural network model effectiveness is sometimes enhanced using the process of transfer learning, as described in [12] as the procedure of pre-training a model within a large dataset and then fine-tuning. The practical aspects of this process, including the initialization of new layers, are described in studies on, for example, fruit detection [13]. The process of feature map extraction and manipulation is crucial here, as highlighted in the analysis of the evolution of YOLO models [14]. This approach has been successfully applied, among others, in the classification of cultural heritage images [11]. However, as research on data augmentation [15] indicates, these models depend on large data sets to avoid overfitting [15]. For this purpose, augmentation techniques are used, e.g., random rotations [15], and, as noted in [16], data normalization for input standardization. Evaluating model performance is crucial, and image recognition is one of the most important fields of computer vision, as confirmed by [17]. Evaluation standards and datasets such as the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) [18] play an important role here. Various metrics are used in these competitions, e.g., top-5 accuracy [18]. To evaluate the trade-off between precision and sensitivity in detection, as indicated in the review of YOLO architectures [19], the Average Precision (AP) index based on the precision-recall curve is used. The literature review confirms significant progress in the field of automatic image analysis, driven by the development of deep neural networks, including convolutional architectures (CNN). Key approaches to object detection, such as models from the YOLO family, and image classification, where ResNet architectures stand out, have been identified. The importance of supporting techniques, such as transfer learning and data augmentation, which are particularly valuable in the context of specialized datasets such as cultural heritage images, has also been discussed. Established evaluation methods and available benchmarks allow for an objective assessment of the developed solutions. To provide the research with the appropriate context, this study focuses on four periods: Romanesque, Gothic, Renaissance, and Baroque. Each of these periods is characterized by a distinct visual language, whose influences were transnational in nature. The selection of France, Italy, Spain, and Poland as the geographical area made it possible to gather a rich and diverse set of data. The scope of the study, as defined above, ensures that the symbols and styles analyzed are representative of typical cultural heritage sites throughout the European Union, which increases the potential for practical application of the obtained results. All this information allowed to formulate the following thesis and hypotheses. Thesis: The use of the YOLOv8 and ResNet50 models to analyze symbols found on cultural monuments allows for the effective classification of their country of origin and historical era. H1. Detection of cultural symbols using the YOLOv8 model achieves an F1-score effectiveness of 80% in classifying historical eras.

H2. Classification of cultural symbols using the ResNet50 model achieves an F1-score effectiveness of 80% in classifying historical eras.

2. Materials and methods

The study used defined data sets, environment configuration, and experimental procedures, which were described in detail at each stage of the work, from data collection to model training and evaluation. The study was based on a collection of approximately 3,200 images of historic buildings. The selection of images was based on the possibility of determining the historical era and country of origin based on reliable sources. Buildings characterized by complexity and richness symbolism were selected, as they constituted suitable material for analysis using machine learning. Care was taken to ensure that the dataset evenly represented the four historical periods analyzed (Romanesque, Gothic, Ba-roque, Renaissance) and the four countries of origin (Poland, Spain, France, Italy) in order to minimize the risk of model bias. Each image was assigned to a single era and a single country. The preparation of data for model training was conducted in two ways. For the YOLOv8 model, responsible for building detection and epoch classification, each image was assigned a bound-ing box around the object and a label corresponding to one of four historical epochs; the annotation process was carried out using the Roboflow platform. For the ResNet50 model, which classifies the country of origin, the photos were manually organized into a folder structure, where the name of each folder corresponded to one of four countries; each folder dedicated to a given country contained approximately 800 photos. The prepared datasets were divided into training, validation, and test subsets. For the YOLOv8 model, the division ratios were 80% training data, 13% validation data, and 7% test data, respectively. For the ResNet50 model, the division was 70% training data, 15% validation data, and 15% test data. During the division, the proportions of classes in each subset were maintained. The research was conducted on a workstation equipped with the hardware shown in Table 1.

Table 1: Hardware Specification

Category	Description
processor	11th Gen Intel(R) Core(TM) i7-11700K @3.60GHz
Graphics card	NVIDIA GeForce RTX 3070
RAM	32GB DDR4 3200MHz
Disk drive	SSD NVMe 1 TB
OS	Windows 10 Education 64-bit

For the YOLOv8 model (yolov8s version, pre-trained), the following hyperparameters were used: initial learning rate (lr0) 0.01, final learning rate (lrf) also 0.01 (the learning rate decreased to 1% of the initial value), batch size 16 and 50 training epochs. Warm-up epochs were also used, the input image size was set to 640x640 pixels, the momentum coefficient to 0.937, and weight regularization. Mosaic augmentation was disabled for the last 10 training epochs.

For the ResNet50 model (pre-trained on ImageNet), the following parameters were configured: batch size 16, learning rates 0.001 for the new classifier layer and 0.0001 for the unlocked 'layer4' layers, weight decay 0.0001. A learning rate scheduler (Scheduler LR) was used, reducing the value by a factor of 0.1 every 7 epochs, and an Adam optimizer. The number of epochs was set to 50, with an early stopping mechanism with a patience of 10. Cross-entropy was selected as the loss function, and gradient clipping was set to 1.0. Dropout rates of 30% and 20% were applied to the respective layers in the new classification module. The ResNet50 modification consisted of replacing the last classifica-tion layer with a new module consisting of the following sequence: Drop-out layer, linear layer with ReLU activation, another lin-ear layer. The initial layers of the Res-Net50 model were frozen during training.

The overall course of the experiment included stages from data collection, through its preliminary processing and categorization, division into sets, implementation, and training of YOLOv8 and ResNet50 models, to eval-uation and interpretation of results. The collected images were first checked for damage; defective files were de-leted. They were then assigned country and era labels. The resolution of the images was standardized: for YOLOv8 to 640x640 pixels, and for ResNet50, random cropping from 256 to 224 pixels was used. For the YOLOv8 model, horizontal reflection and mosaic aug-mentation were used (excluding the last 10 epochs). For the ResNet50 model, a wider set of transformations was used, including random horizontal reflection (50% prob-ability), random vertical reflection (20% probabil-ity), random image rotations (maximum rotation angle 15 de-grees, Color Jitter image property modifications (change in brightness, contrast, saturation in the range [0.7, 1.3] and hue in the range [-0.1, 0.1]), random per-spective change (50% probability) and random conver-sion to grayscale (10% probability). For the ResNet50 model, Z-score normalization was used, transforming pixel values so that their mean was zero and their standard deviation was one.

The training process followed the iterative meth-od with GPU acceleration through the CUDA architec-ture. Progress with the training process was tracked with loss functions and metrics for accuracy. The model for Res-Net50 implemented early stopping with a patience of 10 epochs to avoid the occurrence of over-fitting. Each of the models was trained ten times for 50 epochs and the model with the top performance against the validation set chosen for further inspection. The trained models were finally evaluated for separate test sets to which the trained models had access for the first time. Performance metrics like mAP (for varied IoU thresh-olds) were used for the YOLOv8 model. For the two models and for Res-Net50 for the classification task, metrics like accuracy, precision, sensitivity, F1-score, and error matrix were used.

3. Results

The experiments provided data on the effectiveness of the YOLOv8 and ResNet50 models in analyzing cultural rel-ics. The models were trained according to the previously described procedure, implementing ten repetitions of 50 epochs for each model and selecting the best variant. The performance of the YOLOv8 model in the task of detect-ing monuments and classifying historical epochs was de-termined using the metric of average precision (mAP). After 50 epochs of training, the model achieved a mAP50 value (for an Intersection over Union (IoU) threshold of 0.5) of 0.873. For a range of IoU thresholds from 0.5 to 0.95 (mAP50-95), the average precision was 0.689.

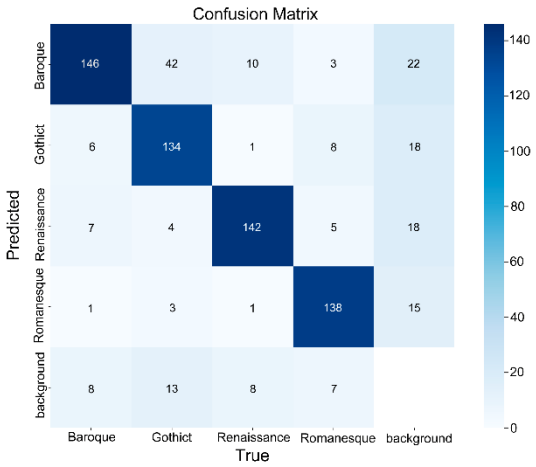

Figure 1: Confusion matrix for YOLOv8 model.

The diagonal of the matrix (Figure 1) represents cor-rectly classified cases. For the Baroque period, 146 paint-ings were correctly classified, for the Gothic period 134, for the Renaissance period 142, and for the Romanesque period 138. The values outside the diagonal indicate in-correct classifications. For example, for the Baroque class, 12 instances were incorrectly assigned to Gothic, 10 to Renaissance, 3 to Romanesque, and 22 to back-ground. Based on the data from the error matrix, preci-sion, sensitivity, and F1-score indicators were calculated.

Table 2: Classification metrics for YOLOv8 model

Epoch	Precision	Recall	F1-score
Baroque	0.85	0.91	0.88
Gothic	0.90	0.88	0.89
Renaissance	0.90	0.92	0.91
Romanesque	0.97	0.90	0.93
Aver-age(macro)	0.90	0.90	0.90

The highest F1-score was recorded for the Romanesque period and reached the value of 0.93 whilst Precision was at 0.97 (Table 2). The next highest F1-score was noted for Renaissance epoch followed by gothic era where the value was 0.89. Baroque era achieved the worst result amounting to F1-score at 0.88 and precision at 0.85. The average macro values for precision, sensitivity, and F1-score were 0.90. On the precision-Recall curve (Figure 2), the average precision for styles was: 0.857 for Baroque, 0.845 for Gothic, 0.861 for Renaissance, and 0.930 for Romanesque. The average precision averaged at an IoU threshold of 0.5 is 0.873. The Romanesque style achieved the highest performance.

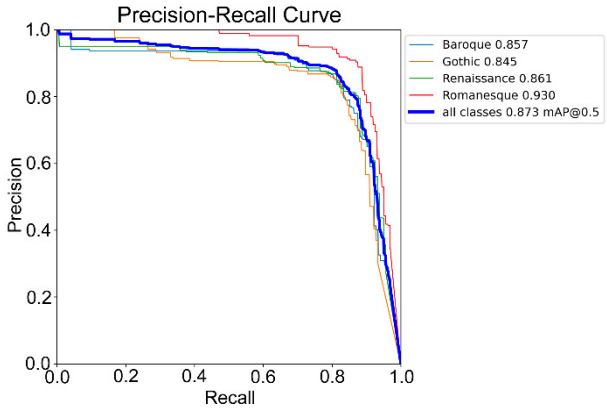

Figure 2: Precision-Recall Curve for YOLOv8 model.

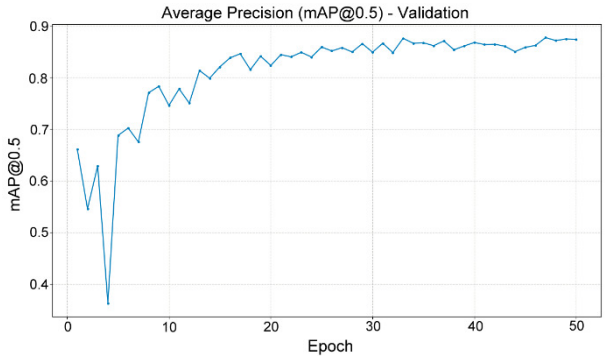

Figure 3: Average Precision for YOLOv8 model on validation set.

The mAP@0.5 on the validation set (Figure 3), after initial volatility and a phase of rapid growth up to approximately 0.83 in epoch 15, stabilizes at a high level, oscillating between 0.86 and 0.88, ending training with a value of approximately 0.875.

Table 3: Classification metrics for ResNet50 model

Class	Precision	Recall	F1-Score	Support
France	0.89	0.72	0.80	119
Spain	0.85	0.80	0.82	120
Poland	0.91	0.94	0.93	123
Italy	0.73	0.88	0.80	119
Macro Avg	0.84	0.84	0.84	481
Weighted Avg	0.85	0.84	0.84	481

The ResNet50 model was used for the task of classifying the country of origin of historic buildings coming from four countries:

- France.
- Spain.
- Poland.
- Italy.

This model achieved an overall accuracy of 84% on the test set. A detailed classification report is presented in Table 3. The model achieved the highest F1-score 0.93 for the Poland class with precision 0.91, recall at 0.94. For the Spain class, the F1-score was 0.82. The France and Italy classes the recorded value was 0.80. The macro average for the F1 score was 0.84. The effectiveness of country classification by the ResNet50 model is visualized by the error matrix.

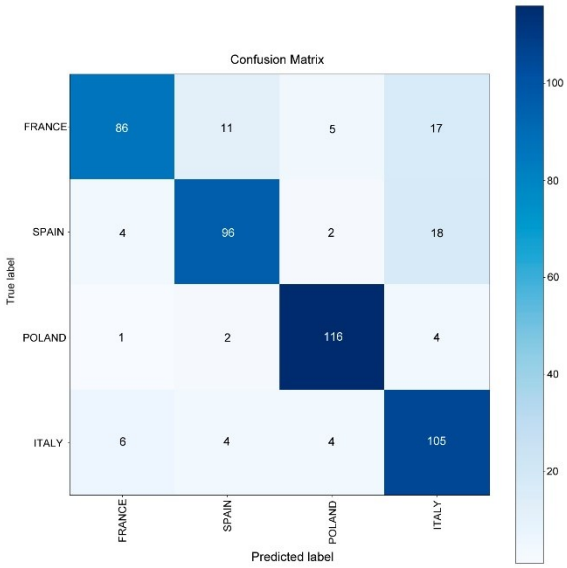

Figure 4: Confusion matrix for ResNet50 model.

Analysis of the confusion matrix (Figure 4) shows that the model correctly classified 86 instances for France, 96 for Spain, 116 for Poland, and 105 for Italy. These counts directly correspond to the recall values for each class, derived from the normalized confusion matrix, which were 0.72 for France, 0.80 for Spain, 0.94 for Poland, and 0.88 for Italy, respectively.

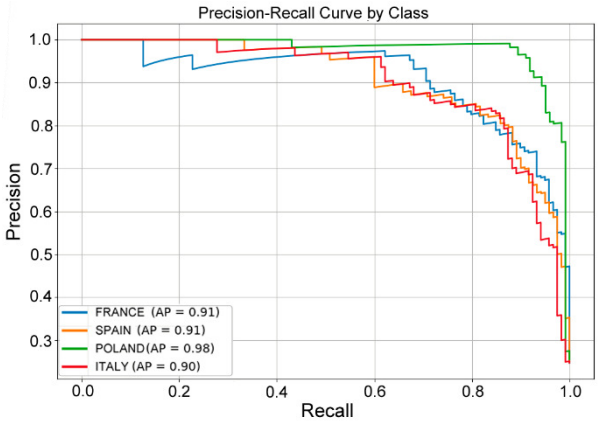

Figure 5: Precision-Recall Curve for ResNet50 model.

The highest AP value 0.98 was obtained for the Poland class. The France and Spain classes obtained an AP of 0.91, while for the Italy class, the AP was 0.90. The loss function on the training set decreases rapidly in the first epochs, reaching values close to 0.2 in the final epochs. The loss function on the validation set also decreases in the initial epochs, then shows greater fluctuations. The accuracy of the training set increases rapidly, reaching values close to 0.95 in the final epochs. The accuracy on the validation set increases, oscillating around 0.80-0.85 in the later phase of training.

Figure 6: Example predictions of architectural epochs by the YOLOv8 model on validation images.

4. Discussion

The YOLOv8 model, used for the detection and classification of historical periods, achieved a high level of effectiveness, with an average mAP50 precision of 0.873. These results demonstrate the model's good overall ability to correctly locate objects. An analysis of the classification of eras, which are based on metrics from Table 1 showed that the Romanesque style achieved the highest F1-score of 0.93. This may be due to the fact that the visual representations of this era are like each other while still distinctive compared to other styles analyzed. This might made it easier for model to learn and recognize them. Confusion matrix revealed that the model tended to confuse classifications between eras such as Baroque, Gothic and Renaissance. These mistakes can be the product of historical style connections or regional architectural differences that obfuscate definite style lines and make classification problematic. Performance curves like Precision-Recall (Figure 2) further demonstrate the model's high capacity to classify individual periods properly. The trajectory of the loss function and metrics of validation, and the milestone of reaching a score of about 0.875 for the measure of mAP@0.5 in the validation set, indicates that this is a good and stable learning process with no evident symptoms of severe overfitting in the case of the primary detection metrics. In the country-of-origin classification task, the ResNet50 model achieved an overall accuracy of 0.84 on the test set. A detailed classification report (Table 3) that the highest F1-score 0.93 was obtained for Poland. As in the case of the Romanesque style for YOLOv8, this may be due to the presence of more distinctive or less diverse features of Polish architecture in the analyzed dataset, which made it easier for the model to learn. Analysis of the error matrix showed that the most common classification errors involved confusing French architecture with Italian and Spanish architecture, as well as Spanish architecture with Italian architecture. The course of the Res-Net50 model training process revealed rapid improvement on the training set. However, the stabilization and even a slight increase in validation loss observed while the training loss continued to decrease in later epochs may suggest a certain degree of model overfitting.

5. Conclusions

From the conducted research and analysis of the obtained results, conclusions as elaborated below are drawn. The achieved metrics of the YOLOv8 model, with the average macro F1-score of 0.90 for the classification of historical periods, confirm hypotheses H1. Similarly, the ResNet50 model, with an average macro F1-score and precision of 0.84 in classifying the country of origin confirmed hypothesis H2. Both models show good performance in their tasks, but there is room for improvement. The models were trained on relatively small dataset which opens possibility for further research using bigger dataset, which could improve generalization and accuracy. Research could also include a larger number of newer models and more advanced augmentation techniques.

References

[1] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You Only Look Once: Unified, Real-Time Object Detection, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016) 779–788, https://doi.org/10.1109/CVPR.2016.91.

[2] T. Diwan, G. Anirudh, J. V. Tembhurne, Object detection using YOLO: challenges, architectural successors, datasets and applications, Multimedia

Tools and Applications 82 (2023) 9243–9275, https://doi.org/10.1007/s11042-022-13644-y.

[3] Chahid, A. Kerkour Elmiad, M. Badaoui, Data Preprocessing For Machine Learning Applications in Healthcare: A Review, Proceedings of the 14th International Conference on Intelligent Systems: Theories and Applications (SITA) (2023) 10373591, https://doi.org/10.1109/SITA60746.2023.10373591.

[4] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going Deeper with Convolutions, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2015) 1–9, https://doi.org/10.1109/CVPR.2015.7298594.

[5] P. Simon, U. V, Deep Learning based Feature Extraction for Texture Classification, Procedia Computer Science 171 (2020) 1680–1687, https://doi.org/10.1016/j.procs.2020.04.180.

[6] M. Krichen, Convolutional Neural Networks: A Survey, Computers 12 (2023) 151, https://doi.org/10.3390/computers12080151.

[7] J. Terven, D.-M. Córdova-Esparza, J.-A. Romero-González, A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS, Machine Learning and Knowledge Extraction 5 (2023) 1680–1716, https://doi.org/10.3390/make5040083.

[8] Md. T. Islam, B. M. N. K. Siddique, S. Rahman, T. Jabid, Image Recognition with Deep Learning, Proceedings of the International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS) (2018) 106–110, https://doi.org/10.1109/ICIIBMS.2018.8550031.

[9] Y. Lai, A Comparison of Traditional Machine Learning and Deep Learning in Image Recognition, Journal of Physics: Conference Series 1314 (2019) 012148, https://doi.org/10.1088/1742-6596/1314/1/012148.

[10] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, Journal of Machine Learning Research 21 (2020) 1–67, http://jmlr.org/papers/v21/20-074.html.

[11] R. Varghese, S. M, YOLOv8: A Novel Object Detection Algorithm with Enhanced Performance and Robustness, Proceedings of the 2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems (ADICS) (2024) 10533619, https://doi.org/10.1109/ADICS58448.2024.10533619.

[12] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, X. Tang, Residual Attention Network for Image Classification, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017) 6450–6458, https://doi.org/10.1109/CVPR.2017.683.

[13] M. Hussain, YOLO-v1 to YOLO-v8, the Rise of YOLO and Its Complementary Nature toward Digital Manufacturing and Industrial Defect Detection, Machines 11 (2023) 677, https://doi.org/10.3390/machines11070677.

[14] K. P. Ferentinos, Deep learning models for plant disease detection and diagnosis, Computers and Electronics in Agriculture 145 (2018) 311–318, https://doi.org/10.1016/j.compag.2018.01.009.

[15] C. Shorten, T. M. Khoshgoftaar, A survey on Image Data Augmentation for Deep Learning, Journal of Big Data 6 (2019) 60, https://doi.org/10.1186/s40537-019-0197-0.

[16] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444, https://doi.org/10.1038/nature14539.

[17] K. He, X. Zhang, S. Ren, J. Sun, Deep Residual Learning for Image Recognition, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016) 770–778, https://doi.org/10.1109/CVPR.2016.90.

[18] Sa, Z. Ge, F. Dayoub, B. Upcroft, T. Perez, C. McCool, DeepFruits: A Fruit Detection System Using Deep Neural Networks, Sensors 16 (2016) 1222, https://doi.org/10.3390/s16081222.

[19] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, L. Fei-Fei, ImageNet Large Scale Visual Recognition Challenge, International Journal of Computer Vision 115 (2015) 211–252, https://doi.org/10.1007/s11263-015-0816-y.