INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2024, VOL. 70, NO. 4, PP. 823-829
Manuscript received August 19, 2022; revised October 2024. doi: 10.24425/ijet.2024.152066

# Comparison of Deep Learning approaches in classification of lacial landforms

Paweł Nadachowski, Zbigniew Łubniewski, Karolina Trzcińska, and Jarosław Tęgowski

***Abstract*—Glacial landforms, created by the continuous movements of glaciers over millennia, are crucial topics in geomorphological research. Their systematic analysis affords invaluable insights into past climatic oscillations and augments understanding of long-term climate change dynamics. The classification of these types of terrain traditionally depends on labor-intensive manual or semi-automated methods. However, the emergence of automated techniques driven by deep learning and neural networks holds promise for enhancing efficiency of terrain classification workflows. This study evaluated the effectiveness of Convolutional Neural Network (CNN) architectures, particularly Residual Neural Network (ResNet) and VGG in comparison with Vision Transformer (ViT) architecture in the glacial landform classification task. By using preprocessed input data from Digital Elevation Model (DEM) which covers regions such as the Lubawa Upland and Gardno-Leba Plain in Poland, as well as the Elise Glacier in Svalbard, Norway, comprehensive assessments of those methods were conducted. The final results highlight the unique ability of deep learning methods to accurately classify glacial landforms. Classification process presented in this study can be the efficient, repeatable and fast solution for automatic terrain classification.**

***Keywords*—Convolutional Neural Network (CNN); deep learning; Digital Elevation Model (DEM); Elise glacier; Gardno-Leba Plain; glacial landforms; Lubawa Upland; Residual Neural Network (ResNet); supervised classification; Svalbard; VGG; Vision Transformer (ViT)**

## I. Introduction

GLACIAL landforms, sculpted by the dynamic forces of glaciers thousands of years ago, offer valuable insights into past and present geological processes. In the field of geomorphology, the study of Earth's landforms and the processes that shape them lies a critical task: the classification of these terrain types. Until now, classification has traditionally relied on manual or semi-automatic methods, often prone to time-consuming processes and interpretation errors. However, recent advancements in remote sensing technologies, particularly the availability of high-resolution digital elevation models (DEMs), offer new opportunities for automating and refining the classification process.

The availability of DEM data presents an opportunity to take advantage of the latest advances in deep neural networks. Deep neural networks are currently solving tasks in a variety of fields, particularly in Computer Vision, from image classification to image generation. By using appropriate preprocessing techniques, DEM data can be effectively treated as image data, enabling the use of current deep neural network architectures to classify glacial landforms. This study delves into the exploration of several deep learning architectures. These range from classic Convolutional Neural Network (CNN) architectures, such as ResNet and VGG, to newer architectures incorporating the recently popular attention mechanism, such as the Vision Transformer.

## II. Related Works

Methodologies for determining geomorphological structures have undergone significant evolution over time. From the laborious and subjective manual delineation processes [1], [2] of Earth's surface features to methods based on bathymetry [3], [4], satellite [5] or radar imagery [6], [7]. Recently there has been a transition towards more efficient and automated machine learning methodologies [8]. These modern approaches afford the advantages of objectivity, consistency, and repeatability in interpretation [9]. However, automated techniques such as object-based image analysis (OBIA) are also gradually gaining attention [10].

This study is a continuation of previous research efforts, in particular the study described in [10], in which glacial landforms were classified using a combination of geomorphometric and spectral features. That earlier work was conducted jointly by the same research team and used classical machine learning models such as Random Forest [11] and Support Vector Machine (SVM) [12].

The purpose of this research is to present deep neural network architectures as another novel approach to characterizing terrain types and, in particular, to classifying glacial landforms.

## III. Study Sites

The study sites consist of three different locations with one in Svalbard, Norway and two in northern Poland. They are represented in datasets as digital elevation model (DEM) format, each annotated with labeled types of glacial landforms and different DEM resolutions. The datasets used in this study follow the datasets described in [10], providing detailed information on the data acquisition methodology.

The area surrounding the Elise Glacier in the Kaffiøyra region of Svalbard, Norway, is the first of these study locations. The foreland of the Elise Glacier (shown in Fig. 1) indicates the

Paweł Nadachowski is with Gdansk University of Technology, Gdansk, Poland (e-mail: s170633@student.pg.edu.pl).
Zbigniew Łubniewski is with Gdansk University of Technology, Gdansk, Poland (e-mail: lubniew@eti.pg.edu.pl).
Karolina Trzcińska is with University of Gdansk, Gdansk, Poland (e-mail: karolina.trzcinska@ug.edu.pl).
Jarosław Tęgowski is with University of Gdansk, Gdansk, Poland (e-mail: jaroslaw.tegowski@ug.edu.pl).

retreat of the Elise Glacier from its maximum extent during the Little Ice Age, a period that lasted until the early 1920s [13]. Consequently, meaning that these are fresh glacial areas with clearly preserved glacial and fluvioglacial relief features. The DEM data for the Elise Glacier were obtained from ArcticDEM and are derived from images taken by the DigitalGlobe constellation consisting of the WorldView-1, WorldView-2 and WorldView-3 satellites [14]. The data were downloaded as 32-bit GeoTIFF files with a spatial resolution of 2 meters, derived from mosaic elevation data. Terrain types covering the Elise Glacier region include end moraines, hummocky moraines, outwash and till plains.

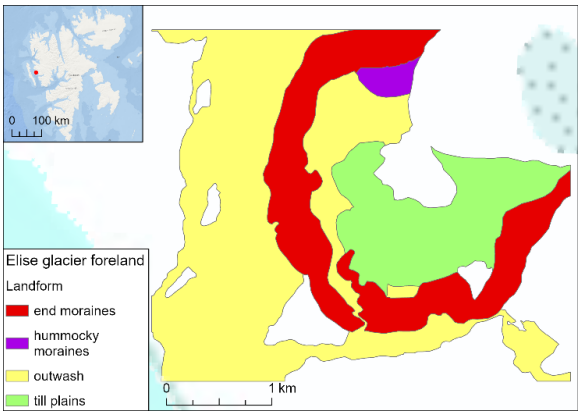


Fig. 1. Location and terrain types of the first study site – foreland of the Elise Glacier (Kaffiøyra region, Svalbard, Norway).

Two additional study sites situated near the Baltic coast in northern Poland are the Gardno-Leba Plain (shown in Fig. 2) and the Lubawa Upland (shown in Fig. 3). These regions were affected by Pleistocene glaciation during the last Glacial Maximum - the Scandinavian ice sheet. Elevation data for these locations were sourced from the database maintained by the Head Office of Geodesy and Cartography in Poland (GUGiK) and were acquired through LiDAR scanning conducted between 2011 and 2014 [15]. The spatial resolution of the DEM for these regions is 5 meters for the Gardno-Leba Plain and 1 meter for the Lubawa Upland. The Gardno-Leba Plain region comprises various terrain formations, including end moraines, hummocky moraines, outwash/glaciolacustrine plains, till plains, and valleys. On the other hand, the Lubawa Upland region includes hummocky moraines, kettle holes, till plains, and valleys.

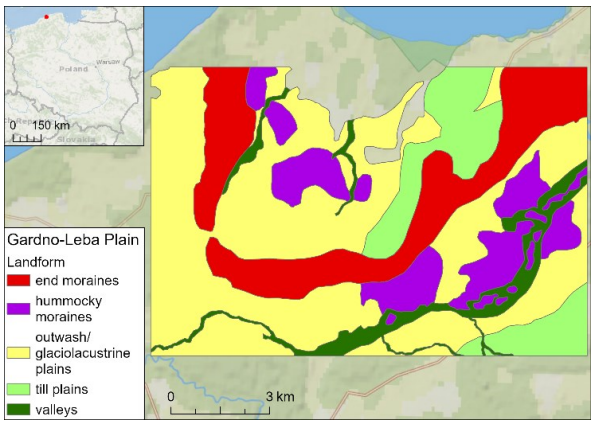


Fig. 2. Location and terrain types of the second study site – Gardno-Leba Plain (northern Poland).

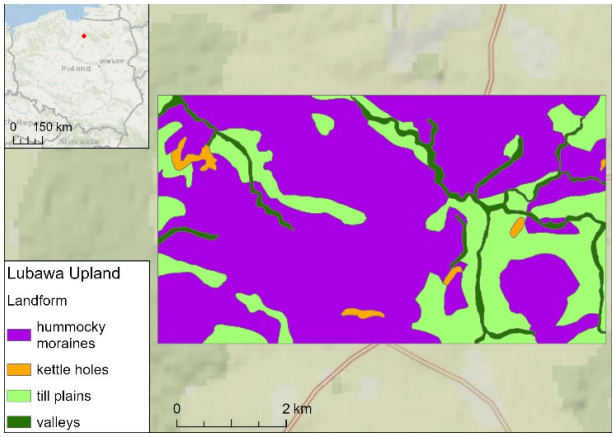


Fig. 3. Location and terrain types of the third study site – Lubawa Upland (northern Poland).

The ground truth labels for terrain types in each study site were constructed by visually inspecting DEMs, existing geologic and geomorphologic maps, orthophotos and terrain identification.

## IV. Deep Learning Architectures

Artificial Neural Networks, particularly Deep Neural Networks (DNNs), have become fundamental in modern artificial intelligence research due to their ability to learn complex patterns and make accurate predictions in a variety of domains [16], [17]. With millions of learnable parameters and multiple layers, DNNs excel at analyzing vast and complex data sets, although they require careful consideration to avoid issues such as over-fitting [18]. Given the grid-like nature of input data in glacial landform classification, networks such as Convolutional Neural Networks (CNNs), in particular the VGG and ResNet architectures are particularly well-suited for this task. Additionally, the study explores the Vision Transformer (ViT), a recent advancement in neural network architectures that has shown promise in handling grid-like data structures. By comparing these different approaches, the research aims to identify the most effective techniques for accurate and efficient glacial landform classification.

### *A. Convolutional Neural Network (CNN)*

Convolutional Neural Networks are predominant in computer vision for their adeptness in extracting meaningful features from data [19]. They comprise convolutional layers, pooling layers, and fully connected layers, enabling hierarchical feature learning from input matrices like images or like in this example DEM data. An example of CNN architecture is shown in Fig. 4. By leveraging the convolutional layers, CNNs can detect distinctive features across layers, capturing simple edges in initial layers and textures or shapes in subsequent ones. Pooling layers further down-sample feature maps while preserving valuable information, aiding in feature extraction and controlling overfitting. Fully connected layers aggregate spatial information from previous layers, facilitating high-level feature representation and classification, making CNNs a promising tool for classifying glacial landforms using DEM data. In this study two types of CNN architectures were used during the experimentations: VGG and Residual Neural Network (ResNet).

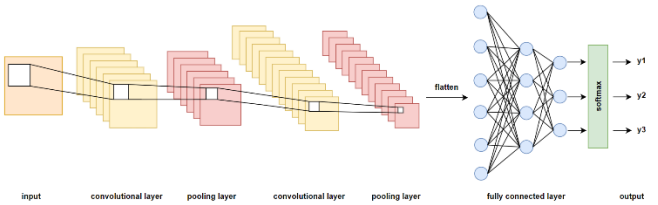


Fig. 4. Visualization of sample CNN architecture

## B. VGG

The VGG network, introduced by the Visual Geometry Group at the University of Oxford in 2014, stands as a prominent CNN architecture [20]. It gained acclaim for its simplicity and uniformity, comprising multiple convolutional layers followed by max pooling layers for down-sampling, and ending with fully connected layers for classification. Utilizing 3 x 3 convolution filters throughout, VGG achieves depth by stacking multiple convolution layers, leading to improved performance in image classification tasks. In practice two VGG types of architectures are used with 16 and 19 layers (convolutional and fully connected) each. These are shown in Fig. 5.

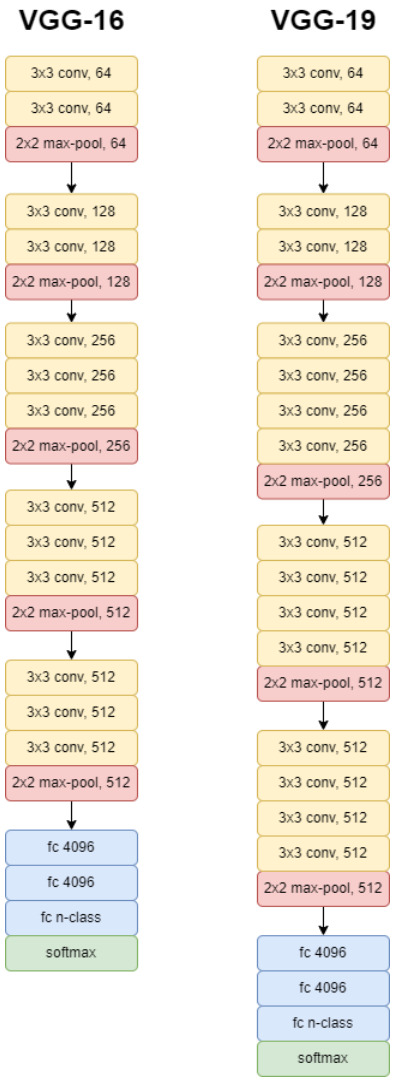


Fig. 5. Visualization of the VGG-16 and VGG-19 architectures

## C. Residual Neural Network (ResNet)

Residual Neural Networks (ResNets) [21] were developed to address the issue of vanishing gradients encountered by very deep neural networks [22]. They introduce residual connections, allowing for more direct information flow through the network. By focusing on learning the difference between input and desired output, ResNets enable the network to prioritize learning subtle changes rather than starting from scratch at each layer. ResNets come in various depths, denoted by numbers like ResNet-50, ResNet-101 or ResNet-152 (which are shown in Fig. 6), with deeper architectures typically achieving better performance but requiring more computational resources and data for training. These networks have shown effectiveness in preserving essential information and mitigating vanishing gradient problems, making them valuable tools for various deep learning tasks.

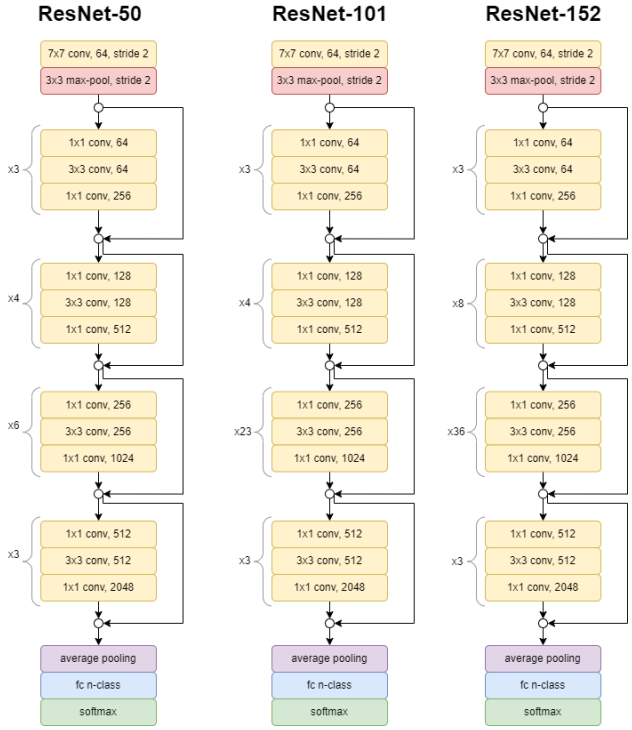


Fig. 6. Visualization of the ResNet-50, ResNet-101 and ResNet-152 architectures

## D. Vision Transformer (ViT)

Vision Transformer (ViT) [23] differs from traditional Convolutional Neural Network architectures by adopting self-attention mechanism originally designed for natural language processing [24]. Attention mechanism in ViT allows this architecture to selectively focus on relevant parts of input data, enhancing its ability to learn and process information effectively. ViT can capture both local and global features within images. Operating directly on fixed-size image patches ViT eliminates the need for handcrafted features and demonstrates remarkable performance across diverse computer vision tasks. The simplified architecture visualization is presented in Fig. 7.

Several modifications have been made to improve the original Vision Transformer architecture. This study used a modification proposed in [25], incorporating Shifted Patch

Tokenization (SPT) and Locality Self-Attention (LSA) mechanisms into the ViT architecture. These improvements enable the model to achieve high performance when learning even on a smaller dataset.

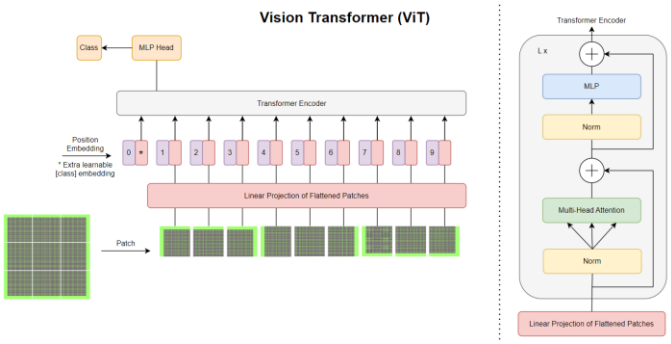


Fig. 7. Visualization of the ViT architecture. Adapted from [23].

## V. METHODS

The digital elevation model (DEM) data utilized in this study consists of X, Y, Z coordinate values. To ensure optimal processing by the neural networks model, it is a need to format the data into a suitable matrix structure. Specifically, the input for the models should be in format of N x M matrix, where N represents the number of samples and M denotes the dimensionality of each sample. Furthermore, each sample must be associated with a corresponding label which indicates the terrain type it represents. Since the CNN and Vision Transformer architectures are made for images, it was decided to split the DEM areas into squares. Then, for each square the elevation points were connected to their terrain type labels for further analysis.

### *A. Data preprocessing, splitting and augmentation*

A random sampling of N points (N=600) per class from the DEM data was conducted for each study site using the "Random selection with subsets" method in QGIS. Subsequently, square polygons were generated at the center of the previously selected points, with dimensions chosen to encompass a 64x64 point area from the DEM. This criterion aimed to ensure adequate coverage of geomorphological features across different resolutions, while maintaining efficiency in model training. The "Buffer" method in QGIS was utilized during point selection to prevent squares from overlapping class boundaries, ensuring that each square contained points exclusively from a single glacial landform class. The resulting linked data was exported as a CSV file and processed in Python using the NumPy and Pandas libraries, yielding data arrays with dimensions Nx64x64, where N represents the sample count. A representative single sample is illustrated in Fig. 8.

Following data preprocessing, the data was partitioned into three subsets: training, validation, and test. The training set, comprising 80% of the data, was utilized for model training. Meanwhile, the validation set, constituting 10% of the data, was employed to optimize hyperparameters and assess model generalization. To mitigate overfitting, the Early Stopping method was implemented, halting training if validation loss failed to improve over a specified number of epochs. Finally, the test set, comprising the remaining 10% of the dataset, was utilized for the conclusive evaluation of model performance.

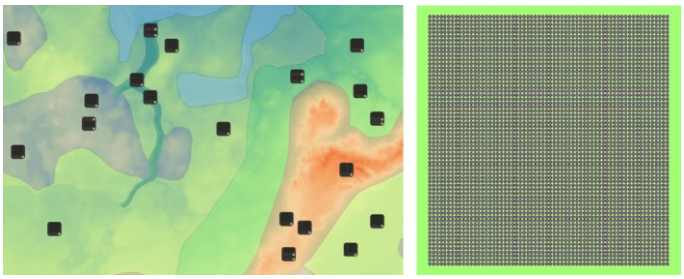

Fig. 8. Visualization of the sampled areas (left) and a zoomed-in view of a single 64x64 sample (right).

To increase the size of the training set for better training of deep learning models, it was decided to use data augmentation methods. Horizontal mirroring, vertical mirroring, and simultaneous horizontal and vertical mirroring were applied to each of the extracted samples, quadrupling the size of the training dataset.

Three different random datasets were generated for each study site. Each went through the same data preparation process. This was aimed to assess the models' ability to learn general patterns in the data and check if they could achieve consistent results regardless of the sampled points from the DEM.

### *B. Model implementation, hyperparameters optimization and training*

Each of the models used during this study was implemented in Python using the PyTorch [26] machine learning framework. The CNN architectures such as VGG-16, VGG-19, ResNet-50, ResNet-101, ResNet-152 were taken from an official package for PyTorch called Torchvision. In order to prepare them for processing data from DEM, they were modified to take 1 channel from each sample instead of 3 as implemented in the original code. In the case of the Vision Transformer model that was used in the experiments, the implementation from the vit-pytorch library [27] was used. This library is based on the PyTorch and has many different modifications to the original ViT architecture implemented.

Before training a machine learning model, setting appropriate hyperparameters is crucial as they do not directly learn but significantly impact training quality. Key hyperparameters in this study included batch size, learning rate, number of epochs, and patience for the Early Stopping method [28]. The batch size, determining the number of samples processed per iteration, was set to 256, ensuring efficient memory usage during training. The learning rate, critical for parameter adjustment, was set to 0.00001 for CNNs, except for the VGG architectures for the Lubawa Upland site where the models performed better with a value of 0.0001 and 0.001 for ViTs, balancing convergence speed and model performance. Additionally, the number of epochs was set to 500, although models rarely reached this limit due to the Early Stopping mechanism with a patience of 50 epochs, preventing overfitting by halting training when classification performance on validation set stopped improving.

The Vision Transformer architecture offers a slightly expanded set of hyperparameters compared to CNNs. These include patch_size, dim, depth, heads, mlp_dim, dropout, and emb_dropout rate, allowing for fine-tuning of model characteristics. Leveraging the Optuna [29] hyperparameter optimization platform, 200 tuning runs were executed per each

site to identify optimal hyperparameter values, aiming to strike a balance between comprehensive exploration of the search space and computational efficiency. The best-performing hyperparameter values for each location, determined by the highest overall validation accuracy, are summarized in Table I.

TABLE I
VIT HYPERPARAMETER VALUES FOR EACH STUDY SITE

| | Elise | Gardno-Leba | Lubawa |
|---|---|---|---|
| **patch_size** | 16 | 8 | 8 |
| **dim** | 64 | 32 | 128 |
| **depth** | 6 | 11 | 9 |
| **heads** | 23 | 11 | 10 |
| **mlp_dim** | 256 | 512 | 32 |
| **dropout** | 0.5 | 0.2 | 0.1 |
| **emb_dropout** | 0.3 | 0.3 | 0.3 |
| **learning_rate** | 0.001 | 0.001 | 0.001 |

Following hyperparameter tuning, models were retrained on each study site. During training, model performance was assessed on a validation set after each epoch to monitor overfitting tendencies. Early Stopping was implemented to halt training upon overfitting detection to prevent further deterioration of model performance and then save the model checkpoint with the highest overall accuracy. Experiment tracking was facilitated using the MLflow [30] library to keep detailed records of each experiment, including configurations, hyperparameters and metrics. The training process was conducted on a machine with a GeForce RTX 3070 Ti graphics card, ensuring smooth and efficient experiment execution.

## VI. RESULTS

After completion of training models, each of them was loaded from the checkpoint and assessed on the test sets. Detailed accuracy assessments for each instance of the models (#1-#3) in each study site are presented in the following tables: Table II for the Elise Glacier site, Table III for the Gardno-Leba Plain site, and Table IV for the Lubawa Upland site.

TABLE II
TEST ACCURACY ASSESSMENT FOR THE ELISE GLACIER STUDY SITE

| Model | #1 | #2 | #3 | Average |
|---|---|---|---|---|
| VGG-16 | 96,3% | 91.3% | 93.3% | 93.6% |
| VGG-19 | **97.9%** | 88.8% | 92.5% | 93.1% |
| ResNet-50 | 97.1% | 96.7% | 95.4% | **96.4%** |
| ResNet-101 | 96.3% | 90.4% | 93.8% | 93.5% |
| ResNet-152 | 95.4% | 93.3% | 90.0% | 92.9% |
| ViT | 97.5% | 92.1% | 94.6% | 94.7% |

TABLE III
TEST ACCURACY ASSESSMENT FOR THE GARDNO-LEBA PLAIN STUDY SITE

| Model | #1 | #2 | #3 | Average |
|---|---|---|---|---|
| VGG-16 | **86.0%** | 84.3% | 82.0% | **84.1%** |
| VGG-19 | 83.7% | 84.7% | 83.3% | 83.9% |
| ResNet-50 | 80.0% | 79.7% | 74.7% | 78.1% |
| ResNet-101 | 73.7% | 83.7% | 77.0% | 78.1% |
| ResNet-152 | 80.3% | 80.7% | 74.7% | 78.6% |
| ViT | 79.3% | 70.7% | 69.0% | 73.0% |

TABLE IV
TEST ACCURACY ASSESSMENT FOR THE LUBAWA UPLAND STUDY SITE

| Model | #1 | #2 | #3 | Average |
|---|---|---|---|---|
| VGG-16 | 54.2% | 55.8% | 54.6% | 54.9% |
| VGG-19 | 54.2% | 52.9% | 53.3% | 53.5% |
| ResNet-50 | 77.9% | **81.7%** | 75.8% | **78.5%** |
| ResNet-101 | 78.8% | 76.3% | 74.6% | 76.6% |
| ResNet-152 | 77.5% | 75.8% | 74.2% | 75.8% |
| ViT | 61.7% | 70.8% | 69.6% | 67.4% |

From the above tables assessing the accuracy of the models, it can be seen that the best results were obtained for the Elise Glacier site. Each of the models achieved more than 90% accuracy for this data. The highest value was achieved by the VGG-19 model in instance #1 with an accuracy of 97.9%. The classification map of the test points along with the confusion matrix is shown in Fig. 9 and Fig. 12. From the confusion matrix, it can be seen that only the till plains and end moraines terrain types have misclassification in a small number of samples. However, averaging all the results from all the experiments, it was the ResNet-50 model that proved to be the best, achieving an average accuracy of 96.4%. The ViT model was second best, with an average accuracy of 94.7%. The ResNet-152 model was the worst, achieving an average accuracy of 92.9%. The differences between all the models are small at a few percent, indicating that the models did quite well in classifying the foreland of the Elise Glacier site.

The results of the experiments on the Gardno-Leba Plain show that the models performed worse than on the Elise Glacier. The range of results extends from 69% to 86% accuracy. The best model was the VGG-16 model achieving a score of 86% accuracy at the #1 instance and the best average score from 3 experiments equal to 84.1% average accuracy. The visualization of classification map and the confusion matrix on the test set are shown in Figure 10 and Figure 13. From the confusion matrix it can be seen that outwash and till plains have most misclassifications. The second-best model was VGG-19, achieving an average accuracy of 84.1%. The worst performing model was ViT, achieving an average accuracy of 73%. It can be noted that in the case of Gardno-Leba Plain site, the usual CNN family models performed much better than ViT architecture.

The models performed worst in the Lubawa Upland site. Here, models such as VGG-16 and VGG-19 were unable to learn relationships between elevation points, achieving a maximum accuracy of 54.2% and very low average accuracy scores. Models from the ResNet architecture fared much better, where ResNet-50 achieved the best result of 81.7% accuracy and 78.5% average accuracy from 3 experiments. The visualization of the classification map for the best instance of model #2 along with the confusion matrix can be observed in Fig. 11 and Fig. 14. The ViT model performed slightly worse than ResNets, achieving results of 67.4% average accuracy.

Summarizing the results obtained from the experiments, it can be seen that CNN models performed significantly better than models with ViT architecture. This may be due to the fact that ViT models require more data for correct classification than CNN models. On the other hand, in terms of the difference between VGG and ResNet models, VGG models perform better on easier maps where the boundaries between different terrain

types are easy to separate, such as the Elise Glacier or the Gardno-Leba Plain. On more difficult maps with more complex land type shapes, such as the Lubawa Upland, they cannot find the relationship between data and terrain labels. This makes networks with ResNet architecture a better choice for the certainty of a high accuracy score during classification for any type of location.

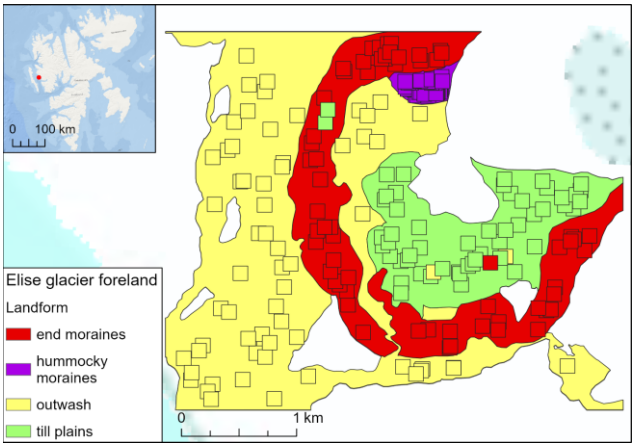


Fig. 9. Visualization of the classification with the VGG-19 best model instance (#1) at the Elise Glacier site.

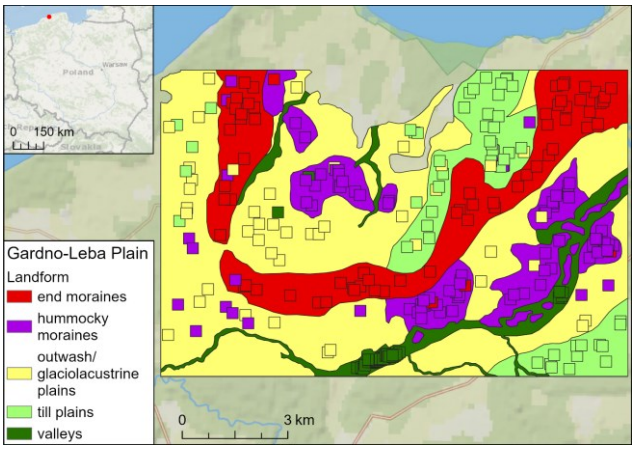


Fig. 10. Visualization of the classification with the VGG-16 best model instance (#1) at the Gardno-Leba Plain site.

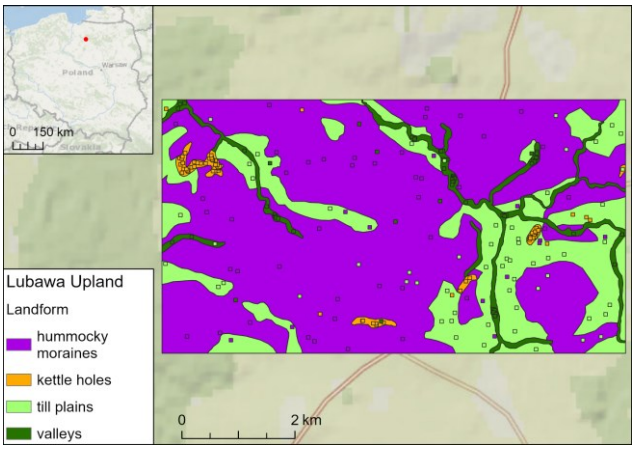


Fig. 11. Visualization of the classification with the ResNet-50 best model instance (#2) at the Lubawa Upland site.

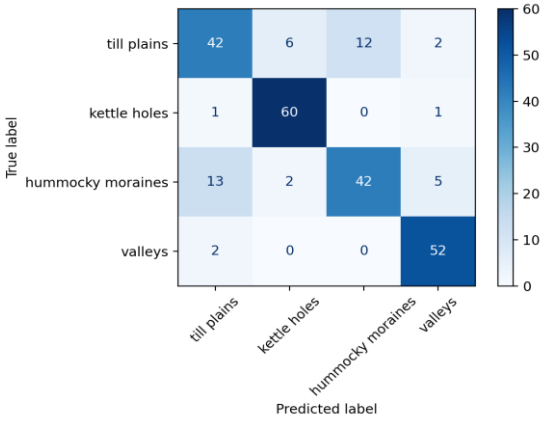


Fig. 12. Confusion matrix for the VGG-19 best model instance (#1) at the Elise Glacier site.

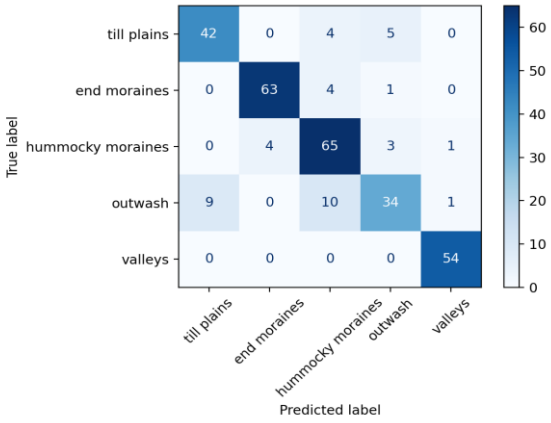


Fig. 13. Confusion matrix for the VGG-16 best model instance (#1) at the Gardno-Leba Plain site.

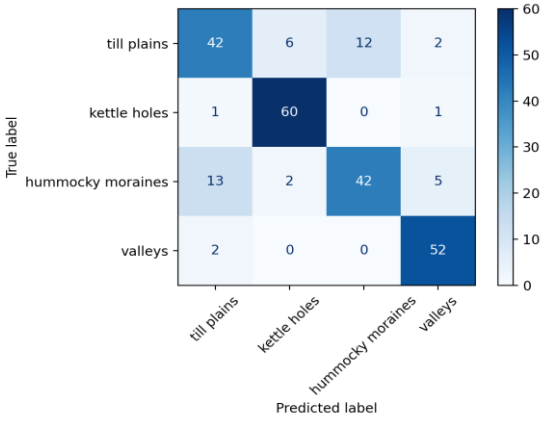


Fig. 14. Confusion matrix for the ResNet-50 best model instance (#2) at the Lubawa Upland site.

## VII. CONCLUSIONS

The study showed that deep neural networks, such as CNN and ViT, can achieve decent results in classifying glacial landforms using only elevation values. The performance of the models of these architectures was demonstrated on three different study sites representing different glacial terrain forms and classification difficulty levels. The experiments also made it possible to compare different network architectures. In this way, it was shown that the classic CNN models perform much

better than the newer ViT models with a fairly small amount of data. Future research should certainly check the behavior of these models on larger datasets and test the classification method described in this study on other types of terrain, not just glacial landforms.

## ACKNOWLEDGEMENTS

The authors would like to thank Aleksandra Malecha-Łysakowska for invaluable contribution to map visualization template used in this research paper.

## REFERENCES

[1] V. H. Brown, C. R. Stokes, and C. O'Cofaigh, "The glacial geomorphology of the north-west sector of the Laurentide ice sheet", J. Maps, vol. 7, no. 1, pp. 409–428, 2011.

[2] B. M. P. Chandler et al., "Glacial geomorphological mapping: A review of approaches and frameworks for best practice", Earth-Sci. Rev., vol. 185, pp. 806–846, 2018.

[3] P. Dunlop, R. Shannon, M. McCabe, R. Quinn, and E. Doyle, "Marine geophysical evidence for ice sheet extension and recession on the Malin shelf: New evidence for the western limits of the British Irish ice sheet", Mar. Geol., vol. 276, nos. 1–4, pp. 86–99, Oct. 2010.

[4] L. R. Bjarnadóttir, M. C. M. Winsborrow, and K. Andreassen, "Deglaciation of the central Barents Sea", Quaternary Sci. Rev., vol. 92, pp. 208–226, 2014.

[5] J. M. Bendle, V. R. Thorndycraft, and A. P. Palmer, "The glacial geomorphology of the Lago Buenos Aires and Lago Pueyrredón ice lobes of central Patagonia", J. Maps, vol. 13, no. 2, pp. 654–673, 2017.

[6] M. Eckerstorfer, H. Ø. Eriksen, L. Rouyet, H. H. Christiansen, T. R. Lauknes, and L. H. Blikra, "Comparison of geomorphological field mapping and 2D-InSAR mapping of periglacial landscape activity at Nordnesfjellet, northern Norway", Earth Surf. Processes Landforms, vol. 43, no. 10, pp. 2147–2156, 2018.

[7] N. Holschuh, K. Christianson, J. Paden, R. B. Alley, and S. Anandakrishnan, "Linking postglacial landscapes to glacier dynamics using swath radar at Thwaites glacier, Antarctica", Geology, vol. 48, no. 3, pp. 268–272, Mar. 2020.

[8] D. C. Mason, T. R. Scott, and H.-J. Wang, "Extraction of tidal channel networks from airborne scanning laser altimetry", ISPRS J. Photogramm. Remote Sens., vol. 61, no. 2, pp. 67–83, 2006.

[9] I. S. Evans, "Geomorphometry and landform mapping: What is a landform?" Geomorphology, vol. 137, no. 1, pp. 94–106, 2012.

[10] L. Janowski, K. Tylmann, K. Trzcinska, S. Rudowski and J. Tegowski, "Exploration of Glacial Landforms by Object-Based Image Analysis and Spectral Parameters of Digital Elevation Model", in IEEE Transactions on Geoscience and Remote Sensing, vol. 60, pp. 1-17, 2022, Art no. 4502817, http://doi.org/10.1109/TGRS.2021.3091771

[11] T. K. Ho, 'Random decision forests', in Proceedings of 3rd international conference on document analysis and recognition, 1995, vol. 1, pp. 278–282.

[12] C. Cortes and V. Vapnik, 'Support-vector networks', Machine learning, vol. 20, no. 3, pp. 273–297, 1995.

[13] K. Tylmann et al., "The local last glacial maximum of the southern Scandinavian ice sheet front: Cosmogenic nuclide dating of erratics in northern Poland", Quaternary Sci. Rev., vol. 219, pp. 36–46, 2019.

[14] C. Porter et al., ArcticDEM. Harvard Dataverse, 2018, http://doi.org/10.7910/DVN/OHHUKH

[15] Head Office of Geodesy and Cartography, GUGIK, Warsaw, Poland, 2017.

[16] Li Deng, Dong Yu, "Deep Learning: Methods and Applications", Now Foundations and Trends, 2014.

[17] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning", Nature, vol. 521, no. 7553. Springer Science and Business Media LLC, pp. 436–444, May 27, 2015. http://doi.org/10.1038/nature14539

[18] X. Ying, "An Overview of Overfitting and its Solutions", Journal of Physics: Conference Series, vol. 1168, no. 2, p. 022022, Feb. 2019.

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", in Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1, Lake Tahoe, Nevada, 2012, pp. 1097–110

[20] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition", in International Conference on Learning Representations, 2015.

[21] 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), 2016, pp. 770–778.

[22] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks", in Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, 13-15 May 2010, vol. 9, pp. 249–256.

[23] A. Dosovitskiy et al., 'An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale', ICLR, 2021.

[24] A. Vaswani et al., 'Attention is All you Need', in Advances in Neural Information Processing Systems, 2017, vol. 30.

[25] S. H. Lee, S. Lee, and B. C. Song, 'Vision Transformer for Small-Size Datasets', arXiv [cs.CV]. 2021.

[26] A. Paszke et al., 'PyTorch: An Imperative Style, High-Performance Deep Learning Library', in Advances in Neural Information Processing Systems 32, Curran Associates, Inc., 2019, pp. 8024–8035.

[27] https://github.com/lucidrains/vit-pytorch

[28] L. Prechelt, "Early Stopping - But When?", Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 55–69, 1998. http://doi.org/10.1007/3-540-49430-8_3

[29] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, 'Optuna: A Next-generation Hyperparameter Optimization Framework', in Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2019.

[30] M. A. Zaharia et al., "Accelerating the Machine Learning Lifecycle with MLflow", IEEE Data Eng. Bull., vol. 41, pp. 39–45, 2018.

INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2024, VOL. 70, NO. 4, PP. 823-829
Manuscript received August 19, 2022; revised October 2024. doi: 10.24425/ijet.2024.152066