Comparison of convolutional networks and transformers for generating 3D-models via binary space partitioning from a single object image

D.N. Gribanov [1], *I.A. Kilbas* [1], *A.V. Mukhin* [1], *R.A. Paringer* [1], *A.V. Kupriyanov* [1,2]

[1] *Samara National Research University, Moskovskoye Shosse 34, Samara, 443086, Russia;*

[2] *Image Processing Systems Institute, NRC "Kurchatov Institute", Molodogvardeyskaya 151, Samara, 443001, Russia*

Abstract

This study explores the use of transformer architecture as an image encoder model for the task of 3D mesh generation from a single image. Traditionally, models based on autoencoder architecture perform such tasks, where an encoder produces a latent representation that a decoder subsequently converts into a 3D model. When processing image-based input, the ResNet18 convolutional network is a commonly used encoder. In this paper, we investigate replacing the convolutional network with a transformer-based approach while using binary space partitioning (BSP) for 3D object generation. Our experiments demonstrate that a transformer-based architecture, specifically the Compact Convolutional Transformer (CCT), can achieve performance comparable to its convolutional counterpart and exceed it both in quantitative metrics and visual quality. The best CCT-based model achieves a Chamfer Distance (CD) of 1.59 and a Light Field Distance (LFD) of 3907, whereas the convolutional variant attains a CD of 1.64 and an LFD of 3981. The CCT-based model also demonstrates superior 3D reconstruction quality on test samples. Additionally, the transformer model requires four times fewer parameters to achieve these results, though computational resources are two times higher in terms of Multiply-Accumulate operations (MACs). These findings indicate that the transformer-based model is more parameter-efficient and can achieve superior results compared to traditional convolutional networks in single-view reconstruction tasks.

Keywords: computer vision, 3D model, neural network, transformer, convolutional network, vector representation, latent vector.

Citation: Gribanov DN, Kilbas IA, Mukhin AV, Paringer RA, Kupriyanov AV. Comparison of convolutional networks and transformers for generating 3D models via binary space partitioning from a single object image. Computer Optics 2025; 49(6): 1247-1252. DOI: 10.18287/COJ1863.

Introduction

There exist various methods for modeling 3D objects, either as mathematical functions or as explicit 3D representations (e.g., meshes) [1 – 7]. Many studies employ an autoencoder architecture to achieve this task. This architecture consists of two main components: an encoder, which generates a latent vector representation of the object, and a decoder, which reconstructs the 3D object from this representation. Autoencoders enable the processing of diverse data types, including images, point clouds, and voxels, as input data. When dealing with images, convolutional networks – most commonly ResNet18 [8] – are typically used as the encoder.

In BSP-Net [5], the authors use an autoencoder where the encoder is ResNet18 [8], while the decoder processes the latent vector and produces the final 3D object or shape by combining partitions from the binary space partitioning (BSP) process. Similar schemes are also used in other works [3, 9], but with different decoder architectures and systems. To further improve BSP-Net, several works have introduced additional enhancements [4, 10], such as the use of Neural Ordinary Differential Equations (NODE) [11] or improved loss functions [4].

In recent years, transformers [12] have been increasingly applied beyond natural language processing (NLP) to computer vision tasks [13 – 17]. Studies have demonstrated that transformers can compete with convolutional networks and often outperform them in terms of memory efficiency and processing speed for certain tasks.

While transformers were originally developed for text processing [12], their application to computer vision was first demonstrated in [14]. This pioneering work introduced the concept of dividing an image into patches, each measuring 16×16 pixels, and processing them using a transformer network to generate the final output. This approach laid the foundation for subsequent research, leading to various modifications and new methodologies for applying transformers to image data [13, 15, 17]. Each successive study proposed refinements to improve performance or adapt transformer models more effectively for vision tasks.

Nevertheless, while the use of transformer architectures in computer vision has shown promise, they still face certain challenges. For instance, in DeepViT [17], the authors observed that increasing the depth of convolutional networks often improves performance, but the same does not necessarily hold for transformer-based architectures. They found that deeper transformer models tend to saturate in performance due to the nature of attention maps. As network depth increases, attention maps across layers become increasingly similar, reducing the model's effectiveness compared to shallower transformers or convolutional networks. To address this issue, the authors proposed a re-attention mechanism, which regenerates attention maps at different layers, enhancing their diversity while maintaining low computational and memory costs.

Other works [15, 18] have found that transformer-based architectures tend to outperform convolutional networks on medium-to-large datasets when trained from scratch. However, on smaller datasets, transformers often underperform unless architectural modifications are introduced. For instance, the Compact Convolutional Transformer (CCT) [15] introduced a modified tokenization method that integrates convolutional layers, along with sequence pooling in the final layers, while retaining most of the original transformer structure from [14].

With these modifications, it is now possible to achieve 90 % accuracy on CIFAR-10 [19] in under 30 minutes of training using an NVIDIA 2080Ti GPU. The authors also conducted experiments on other smaller datasets, demonstrating the superiority of transformer architectures over convolutional networks in such scenarios.

These works demonstrate that transformer-based architectures are continuously becoming more powerful, much like convolutional networks did in earlier years.

This paper explores the feasibility of replacing convolutional networks with transformers for feature extraction and object vector generation in single-view reconstruction tasks. BSP-Net [5] is used as the central 3D model generation framework in our experiments, and the CCT [15] architecture is employed as the transformer-based encoder.

1. Experiment preparation

For our experiments on generating 3D models from a single image, we use the BSP-Net [5] system. This choice is motivated by its strong performance on relevant metrics, the availability of source code and pretrained models, and its suitability for modifications.

BSP-Net is a deep generative network that represents 3D shapes through binary space partitioning, enabling direct generation of compact polygonal meshes. The architecture consists of three main modules that operate on feature vectors extracted by an encoder. First, a multi-layer perceptron produces a set of plane parameters that define implicit equations for spatial subdivision. Second, a grouping operator combines these planes into convex shape primitives through selective neuron connections, where each convex is formed by the intersection of half-spaces defined by the planes. Finally, an assembly layer merges these convex primitives using union operations to reconstruct the complete shape. Unlike methods based on implicit functions that require expensive iso-surfacing procedures, BSP-Net directly outputs watertight polygonal meshes by applying Constructive Solid Geometry operations to extract explicit surfaces from the learned BSP-tree structure.

The primary advantages of BSP-Net lie in its ability to generate compact, low-polygon meshes that preserve sharp geometric features while maintaining computational efficiency during inference. The method produces meshes that are guaranteed to be watertight and can easily be textured or manipulated, addressing a key limitation of voxel-based and implicit function approaches that often result in over-tessellated surfaces. Training is unsupervised, requiring no ground truth convex decompositions, as the network learns to reconstruct shapes using a shared set of convex primitives across the entire training set. This structural consistency naturally establishes part-level correspondence between different shapes. Additionally, the direct mesh generation eliminates the need for iso-surfacing algorithms, reducing inference time to approximately 0.5 seconds per mesh while producing representations that effectively capture both smooth surfaces and sharp edges – a capability that distinguishes it from methods generating only smooth approximations.

All training parameters and configurations follow those specified in BSP-Net [5]. For the dataset, a subset of ShapeNet [20] is used, comprising 13 different object classes and 24 views per object, with an overall count of approximately 40,000 objects. Each single image has a size of 137 pixels by height and width. During training, these images are center-cropped to 128 pixels by height and width. In the current experiments, only the encoder part of the model is trained, while the decoder remains frozen. The encoder model is trained based on latent vectors generated from a voxel encoder model from the BSP-Net work. All models were trained for 1000 epochs.

For the transformer-based model, we employ the Compact Convolutional Transformer (CCT) [15], chosen for its flexibility, open-source implementation, and ease of modification. Several variants of CCT are used in the experiments. We adopt the notation from the original work while introducing additional parameters to distinguish between model configurations. For example, CCT-18/7×2-3-384 consists of 18 transformer encoder layers, a tokenizer with 2 convolutional layers using 7×7 kernels, and an embedding dimensionality of 384. A multiplier of 3 determines the number of hidden neurons per transformer encoder layer (i.e., 3×384 hidden neurons). The following models were created and evaluated: CCT-14/7×1-3-256, CCT-14/7×2-3-256, CCT-28/7×2-4-256, and CCT-18/7×2-4-384.

While this notation captures the key architectural differences, we also introduce a custom variant based on CCT-18/7×2-4-384, referred to as CCT-18/7×2-4-384-v1. The primary modification in this version is the removal of the pooling layer in the second convolutional block of the tokenizer, which increases the output vector size from the tokenizer. Further details on this modification and its impact on performance are discussed in the next section.

For each architecture, Multiply-Accumulate operations (MACs) were calculated to measure the computational complexity of the architecture, as well as two metrics commonly used in 3D modeling: Chamfer Distance (CD) and Light Field Distance (LFD) [21] – to measure the quality of the final generated 3D objects. CD reflects the accuracy of vertex positions, while LFD evaluates the visual similarity between the modeled and original meshes. For both metrics, lower values indicate better performance.

2. Result and evaluation

The training loss for different models is shown in fig. 1 with the first few epochs omitted from the plot due to high initial values. From the figure, we observe that each network converges to different final loss values. For example, ResNet18 achieves the lowest loss value, while CCT-14/7×1-3-256 and CCT-14/7×2-3-256 exhibit the highest. However, further analysis reveals that differences in loss values do not directly correlate with final result quality.

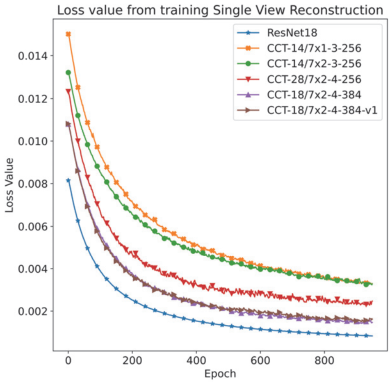

Fig. 1. Training loss during training of different image encoders

The results of the experiments are summarized in tab. 1.

Tab. 1. Evaluation result for BSP-Net system with different image encoders

Image encoder architecture	**Training time, hours**	**Number of parameters, millions**	**MACS, Giga**	**Metric**	
				CD	**LFD**
ResNet18*	-	40	4.7	1.64	3946
ResNet18	**10**	40	4.7	1.68	3981
CCT-14/7×1-3-256	139	**9**	9.4	**1.59**	**3907**
CCT-14/7×2-3-256	13	10	**0.8**	2.21	4116
CCT-28/7×2-4-256	22	23	1.6	1.86	3999
CCT-18/7×2-4-384	28	33	2.3	1.82	4016
CCT-18/7×2-4-384-v1	66	33	8.5	1.82	3976

In the table, we also highlight the ResNet18 model pretrained in the original BSP-Net work, labeled ResNet18*, while another ResNet18 model was trained from scratch to evaluate training time and loss behavior. From the results, we observe that our retrained ResNet18 achieves performance similar to the original BSP-Net version. In the following sections, references to ResNet18 refer to our retrained model.

According to the table, ResNet18 was the fastest to train (10 hours), whereas the CCT architectures required up to 139 hours – nearly 14 times longer. The best-performing model, CCT-14/7×1-3-256, achieved a Chamfer Distance (CD) of 1.59 and a Light Field Distance (LFD) of 3907, which is slightly better than ResNet18. Notably, CCT-14/7×1-3-256 reached this performance while using four times fewer parameters (9 million vs. 40 million for ResNet18), demonstrating the parameter efficiency of the CCT architecture.

However, this conclusion does not apply to all CCT variants. The advantage of CCT-14/7×1-3-256 is likely due to its ability to process more input features, which also increases computational cost. Reducing the number of input features by a factor of 16 in CCT-14/7×2-3-256 speeds up training to the level of ResNet18 but leads to a performance drop. This model achieved a CD of 2.21 and an LFD of 4116 – an increase of 0.62 CD (39% higher) and 209 LFD units compared to CCT-14/7×1-3-256.

We also trained additional architectures with the same number of input features as CCT-14/7×2-3-256: CCT-28/7×2-4-256 and CCT-18/7×2-4-384. These models achieved nearly identical results. For instance, CCT-28/7×2-4-256 achieved a CD of 1.86 and an LFD of 3999 – only 0.27 CD (17 % higher) and 92 LFD units more than CCT-14/7×1-3-256. Although the metric differences were significantly reduced and results improved in these new variants, they still underperformed compared to both CCT-14/7×1-3-256 and ResNet18.

In CCT-18/7×2-4-384-v1, the number of input features was increased by a factor of 4 compared to CCT-18/7×2-4-384, yet there was no significant improvement in the metrics. This may be due to the limited magnitude of the increase in input features or the continued use of two convolutional layers in the tokenizer.

In conclusion, while CCT shows promise as a transformer-based architecture for 3D model generation, these experiments highlight the challenges in identifying the optimal architecture.

Fig. 2 shows visual differences between the trained models on three input samples from the test dataset. The first row presents the input images and the second row represents the source 3D objects in the form of meshes, while the remaining rows display the generated 3D results from each model, as indicated in the first column.

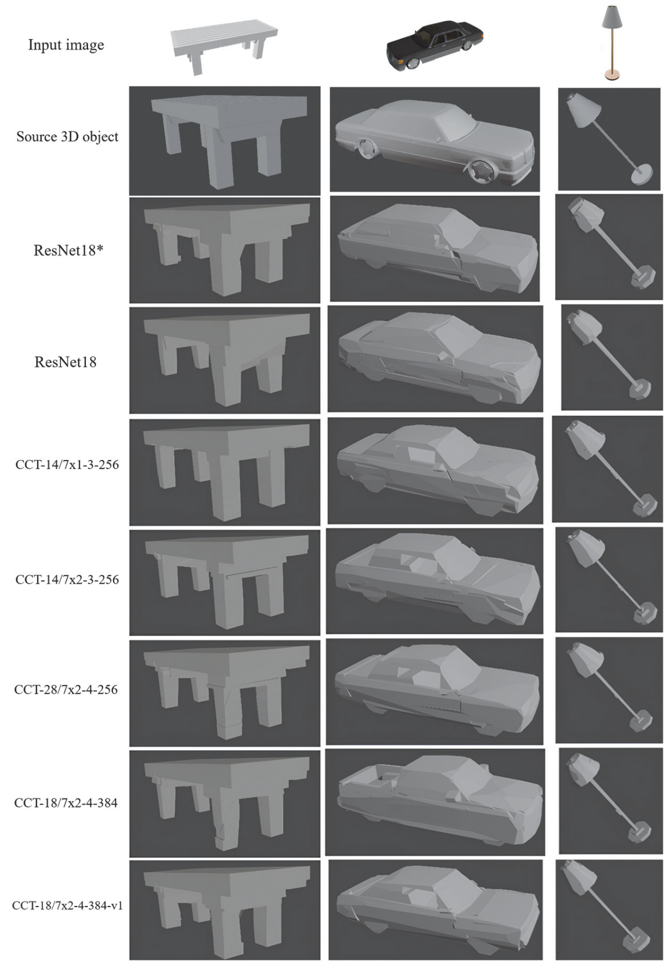

Fig. 2. Generation results from different models on three samples from the test data

From these results, we can observe that nearly every model generates a 3D shape that closely matches the corresponding input image – even CCT-14/7×2-3-256, which has the lowest quantitative metrics.

The first input is a bench, and all models produce a similar 3D object. The main differences lie in the surface quality of the models. For example, the ResNet18-based model produces a bench with extraneous, unnecessary parts on the legs. The clearest and most accurate 3D model is generated by CCT-14/7×1-3-256, which aligns well with its strong metric scores.

The second input is a car. The generated results resemble the input across all models, with one exception: CCT-18/7×2-4-384 produces a different car type than the input, whereas all other models correctly capture the vehicle. Interestingly, the model with the lowest metrics, CCT-14/7×2-3-256, does not exhibit this error.

The final input is a lamp. Results are consistent with those from the first example: differences are mostly visible in the lamp head, while the overall structure remains largely consistent across models.

The combination of quantitative metrics and visual comparisons confirms the potential of transformer architectures in this domain. Notably, CCT-14/7×1-3-256 performs on par with – or even surpasses – the traditional convolutional ResNet18 encoder, both in terms of visual quality and metric performance, while using four times fewer parameters.

These findings suggest strong prospects for future research and improvements. For instance, some studies have proposed distillation techniques between convolutional and transformer-based models using attention mechanisms [22], which result in highly efficient and precise transformer models. This approach could be applied not only between image-trained convolutional and transformer encoders but also with convolutional encoders trained directly on 3D data, such as voxel-based representations. Other research addresses training under data-scarce conditions [23], and we believe such techniques could enhance model robustness and generalization in any training setup.

3. Conclusion

In this paper, we investigated the feasibility of using transformers to generate 3D polygonal meshes from a single object image. The transformer model is used to generate an object vector, which is then passed to a BSP-based decoder to construct the final 3D mesh. As the transformer encoder, we adopted the Compact Convolutional Transformer (CCT), which is well-suited for image-based tasks.

Our experiments demonstrate that CCT-based transformer architectures can deliver comparable – and with certain configurations, superior – results to traditional convolutional networks, while requiring significantly fewer parameters. Several CCT variants were evaluated, including CCT-14/7×1-3-256, CCT-14/7×2-3-256, CCT-28/7×2-4-256, CCT-18/7×2-4-384, and CCT-18/7×2-4-384-v1, to analyze the impact of architectural choices, particularly the number of input features and tokenizer configuration. The best-performing CCT model, CCT-14/7×1-3-256, achieved a Chamfer Distance (CD) of 1.59 and a Light Field Distance (LFD) of 3907, compared to the ResNet18 baseline's CD of 1.64 and LFD of 3981. Visual analysis of test samples further confirmed the quantitative results, showing that CCT-14/7×1-3-256 produces cleaner and more accurate 3D reconstructions compared to ResNet18, with fewer artifacts and better surface quality. This shows that the transformer model not only competes in output quality but also does so with four times fewer parameters, highlighting its parameter efficiency. However, this parameter efficiency comes with increased computational costs, as the CCT model requires approximately two times more Multiply-Accumulate operations (MACs) and significantly longer training times – up to 139 hours compared to 10 hours for ResNet18.

With continued research and refinement, we believe that transformer-based models can achieve even better results. The rapid advancement of transformer techniques has already introduced many simple yet powerful improvements, opening promising directions for future exploration in 3D reconstruction and beyond. For instance, distillation techniques between convolutional and transformer-based models [22] and methods for training under data-scarce conditions [23] could further enhance model performance and robustness. The parameter efficiency and superior quality results compared to CNN models demonstrated by applying the existing CCT architecture to our 3D reconstruction task suggest potential applications in specialized imaging systems, such as single-pixel imaging methods where parameter efficiency and quality is crucial [24, 25]. Furthermore, the compact nature of the CCT architecture makes it particularly suitable for deployment in real-world computer vision systems, including quality inspection applications [26], agricultural monitoring and weed detection [27], structural defect assessment [28], mobile robotics navigation [29], and real-time object detection on resource-constrained devices [30]. These applications would benefit from the parameter efficiency, which correlates with reduced computational requirements, while maintaining the high reconstruction quality that our adapted transformer-based approach demonstrates.

Acknowledgements

The research was carried out within the state assignment theme FSSS-2023-0006.

References

[1] Li A, Zhu Z, Wei M. GenPC: Zero-shot Point Cloud Completion via 3D Generative Priors. arXiv Preprint. 2025. Source: <https://arxiv.org/abs/2502.19896>. DOI: 10.48550/arXiv.2502.19896.

[2] Mo S, Xie E, Chu R, Hong L, Niessner M, Li Z. DiT-3D: Exploring Plain Diffusion Transformers for 3D Shape Generation. Neural Information Processing Systems. 2023; 36: 67960-67971.

[3] Li Y, Dou Y, Chen X, Ni B, Sun Y, Liu Y, Wang F. 3DQD: Generalized Deep 3D Shape Prior via Part-Discretized Diffusion Process, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) 2023: 16784-16794. DOI: 10.1109/CVPR52729.2023.01610.

[4] Gribanov D, Kilbas I, Mukhin A, Paringer R. Effect of Encoder Architectures on the Generation of Vector Representations for Modeling 3D Objects via the Space of Convex Sets. X International Conference on Information Technology and Nanotechnology (ITNT) 2024: 1-7. DOI: 10.1109/itnt60778.2024.10582346.

[5] Feng Y, Tagliasacchi A, Zhang H. BSP-Net: Generating Compact Meshes via Binary Space Partitioning. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) 2020: 42-51. DOI: 10.1109/CVPR42600.2020.00012.

[6] Choy CB, Xu D, Gwak JY, Chen K, Savarese S. 3D-R2N2: A Unified Approach for Single and Multi-view 3D Object Reconstruction. European Conference on Computer Vision (ECCV) 2016: 628-644. DOI: 10.1007/978-3-319-46484-8_38.

[7] Chen R, Yin X, Yang Y, Tong C. Multi-view Pixel2Mesh++: 3D reconstruction via Pixel2Mesh with more images. The Visual Computer. 2022; 39: 5153-5166. DOI: 10.1007/s00371-022-02651-7.

[8] He K, Zhang X, Ren S, Sun J. Identity Mappings in Deep Residual Networks. European Conference on Computer Vision (ECCV) 2016: 630-645. DOI: 10.1007/978-3-319-46493-0_38

[9] Mittal P, Cheng Y, Singh M, Tulsiani S. AutoSDF: Shape Priors for 3D Completion, Reconstruction and Generation. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) 2022: 306-315. DOI: 10.1109/cvpr52688.2022.00040.

[10] Hui KC, Li R, Hu J, Fu C. Neural Template: Topology-aware Reconstruction and Disentangled Generation of 3D Meshes. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) 2022: 18551-18561. DOI: 10.1109/CVPR52688.2022.01802.

[11] Gupta K, Chandraker M. Neural Mesh Flow: 3D Manifold Mesh Generation via Diffeomorphic Flows. Neural Information Processing Systems. 2020; 33: 1747-1758.

[12] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A, Kaiser Ł, Polosukhin I. Attention Is All You Need. Neural Information Processing Systems. 2017; 30: 5998-6008.

[13] Koner R, Jain G, Jain P, Tresp V, Paul S. LookupViT: Compressing visual information to a limited number of tokens. European Conference on Computer Vision 2024: 322-337. DOI: 10.1007/978-3-031-73016-0_19.

[14] Dosovitskiy A, Beyer L, Kolesnikov A, Weissenborn D, Zhai X, Unterthiner T, Dehghani M, Minderer M, Heigold G, Gelly S, Uszkoreit J, Houlsby N. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. International Conference on Learning Representations 2021.

[15] Hassani A, Walton S, Shah N, Abuduweili A, Li J, Shi H. Escaping the Big Data Paradigm with Compact Transformers. arXiv Preprint. 2021. Source: <https://arxiv.org/abs/2104.05704>. DOI: 10.48550/arXiv.2104.05704.

[16] Chandra AA, Tünnermann L, Löfstedt T, Grätz R. Transformer-based deep learning for predicting protein properties in the life sciences. eLife. 2023; 12. DOI: 10.7554/elife.82819.

[17] Zhou D, Kang B, Jin X, Yang L, Lian X, Jiang Z, Hou Q, Feng J. DeepViT: Towards Deeper Vision Transformer. arXiv Preprint. 2021. Source: <https://arxiv.org/abs/2103.11886>. DOI: 10.48550/arxiv.2103.11886.

[18] Yuan L, Chen Y, Wang T, Yu W, Shi Y, Jiang Z, Tay F, Feng J, Yan S. Tokens-to-Token ViT: Training Vision Transformers from Scratch on ImageNet. IEEE/CVF International Conference on Computer Vision (ICCV) 2021: 538-547. DOI: 10.1109/ICCV48922.2021.00060.

[19] Krizhevsky A. Learning Multiple Layers of Features from Tiny Images. 2009.

[20] Chang AX, Funkhouser T, Guibas L. et al. ShapeNet: An Information-Rich 3D Model Repository. arXiv Preprint. 2015. Source: <https://arxiv.org/abs/1512.03012>. DOI: 10.48550/arXiv.1512.03012.

[21] Chen D, Tian X, Shen Y, Ouhyoung M. On Visual Similarity Based 3D Model Retrieval. Computer Graphics Forum. 2003; 22(3): 223-232. DOI: 10.1111/1467-8659.00669.

[22] Touvron H, Cord M, Douze M, Massa F, Sablayrolles A, Jégou H. Training data-efficient image transformers & distillation through attention. International Conference on Machine Learning 2021. 139: 10347-10357.

[23] Lee SH, Lee S, Song BC. Vision Transformer for Small-Size Datasets. arXiv Preprint. 2021. Source: <https://arxiv.org/abs/2112.13492>. DOI: 10.48550/arxiv.2112.13492.

[24] Babukhin DV, Reutov AA, Sych DV. Study of image reconstruction efficiency in single-pixel imaging method using generative adversarial networks. Computer Optics 2025; 49(5): 818-825. DOI: 10.18287/2412-6179-CO-1526.

[25] Reutov AA, Babukhin DV, Sych DV. Object classification using a single-pixel camera and neural networks. Computer Optics 2025; 49(3): 517-524. DOI: 10.18287/2412-6179-CO-1538.)

[26] Ndukwe IK, Yunovidov D, Bahrami MR, Mazzara M, Olugbade TO. Quality inspection of fertilizer granules using computer vision - a review. Computer Optics 2025; 49(1): 84-94. DOI: 10.18287/2412-6179-CO-1458.

[27] Shadrin D, Illarionova S, Kasatov R, Akimenkova M, Rudensky G, Erhan E. Weed detection on embedded systems using computer vision algorithms. Computer Optics 2025; 49(1): 103-111. DOI: 10.18287/2412-6179-CO-1454.

[28] Suetin MN, Dementiev VE, Tashlinskii AG, Magdeev RG. Methodology for detecting and assessing the dynamics of defects in engineering structures by processing images from an unmanned aerial vehicle. Computer Optics 2024; 48(5): 762-771. DOI: 10.18287/2412-6179-CO-1438.

[29] Belkin IV, Abrameko AA, Bezuglyi VD, Yudin DA. Localization of mobile robot in prior 3D LiDAR maps using stereo image sequence. Computer Optics 2024; 48(3): 406-417. DOI: 10.18287/2412-6179-CO-1369.

[30] Zagitov A, Chebotareva E, Toschev A, Magid E. Comparative analysis of neural network models performance on low-power devices for a real-time object detection task. Computer Optics 2024; 48 (2): 242-252. DOI: 10.18287/2412-6179-CO-1343).

About authors

Danil Nikolaevich Gribanov (b. 2000), master's student of the Faculty of Informatics at Samara National Research University. Postgraduate student at Samara National Research University. Research assistant of the research laboratory for Automated Scientific Research Systems. Research interests include data mining, computer vision and artificial neural networks. E-mail: *gribanov.dn@ssau.ru*

Igor Alexandrovich Kilbas (b. 2000), master's student of the Faculty of Informatics at Samara National Research University. Postgraduate student at Samara National Research University. Research assistant of the research laboratory for Automated Scientific Research Systems and research laboratory “Artificial intelligence in manufacturing systems”. Research interests include large language models, artificial neural networks and language models. E-mail: *kilbas.ia@ssau.ru*

Artem Vladimirovich Mukhin (b. 1999), master's student of the Faculty of Informatics at Samara National Research University. Postgraduate student at Samara National Research University. Research assistant of the research laboratory for Automated Scientific Research Systems and research laboratory “Artificial intelligence in manufacturing systems”. Research interests include data mining, computer vision, artificial neural networks and real-time high-load systems. E-mail: *mukhin.av@ssau.ru*

Rustam Alexandrovich Paringer (b. 1990), received Master’s degree in Applied Mathematics and Informatics from Samara State Aerospace University (2013). He received his PhD in 2017. Associate professor of the Technical Cybernetics department of Samara National Research University. Research interests: data mining, machine learning and artificial neural networks. E-mail: *RusParinger@ssau.ru*

Alexander Viktorovich Kupriyanov (b. 1978), graduated with honors from Samara State Aerospace University (SSAU) (2001). Candidate’s degree in Technical Sciences (2004) and Doctor of Engineering Science (2013). Currently, Senior Researcher at the Image Processing Systems Institute, Russian Academy of Sciences, head of the Department of Technical Cybernetics at SSAU’s and director of the Institute of IT, Mathematics and Electronics at SSAU’s. Areas of interest: digital signals and image processing, pattern recognition and artificial intelligence, nanoscale image analysis and understanding, biomedical imaging and analysis. More than 90 scientific papers, including 42 published articles and 2 monographs. E-mail: *akupr@ssau.ru*

Received October 17, 2025. The final version – December 06, 2025.