

Научная статья
УДК 721(470.11)
EDN: QQZRBY
DOI: 10.21285/2227-2917-2025-1-155-164



Анализ технических возможностей развития последовательно обучающихся нейронных сетей в проектировании промышленной архитектуры

К.А. Меркушев

Южно-Уральский государственный университет, Челябинск, Россия

Аннотация. Цель исследования заключается в анализе механических возможностей развития последовательно обучающихся нейронных сетей в контексте проектирования промышленной архитектуры. В ходе исследования был выявлен дефицит эмпирических данных по применению нейросетевых методов для автоматизации создания архитектурных моделей промышленных объектов в современной информационной среде. Существующие работы ограничиваются общим описанием проблематики и не предлагают комплексных решений. Применяемые методы основаны на обучении нейронных сетей с использованием изображений, но адаптированы для моделирования автономного поведения в архитектурном проектировании. Предложенные стандарты предусматривают, что алгоритмически управляемые нейронные сети выполняют роль инструмента, облегчающего последующее проектирование, а не являются источником знаний. Основным методом исследования – проведение теоретических и экспериментальных исследований, направленных на разработку новых технологий инженерного дизайна для создания модели автономного поведения в архитектуре. Результаты демонстрируют, что интеграция инженерного дизайна с нейросетевыми методами способствует стандартизации проектных процессов, повышению функциональной эффективности и качества реализуемых объектов. В заключение подтверждено, что применение разработанных методов позволяет на ранних этапах проектирования перейти от традиционных визуальных подходов к использованию алгоритмически управляемых моделей, что открывает новые перспективы в автоматизации инженерного проектирования.

Ключевые слова: автономное проектирование, деградация последовательности, промышленная архитектура, нейронные сети, инженерный дизайн, гибридная архитектура

Для цитирования: Меркушев К.А. Анализ технических возможностей развития последовательно обучающихся нейронных сетей в проектировании промышленной архитектуры // Известия вузов. Инвестиции. Строительство. Недвижимость. 2025. Т. 15. № 1. С. 155–164. <https://doi.org/10.21285/2227-2917-2025-1-155-164>. EDN: QQZRBY.

Original article

Technical capacity for development of sequentially training neural networks in industrial architecture design

Konstantin A. Merkushev

South Ural State University, Chelyabinsk, Russia

Abstract. This paper aims to analyze the possibilities of developing sequentially learning neural networks in the context of industrial architecture design. The study revealed a lack of empirical data on the application of neural network methods to automate the creation of architectural models of industrial objects in the modern information environment. Moreover, available studies are limited to a general description of the problem, offering no comprehensive solutions. The applied methods are based on training neural networks using images; however, they are adapted for modeling autonomous behavior in architectural design. The proposed standards stipulate that algorithmically controlled neural networks serve as a tool to facilitate subsequent design, rather than being a source of knowledge. Theoretical and experimental studies have been conducted to develop new engineering design techniques to create a model of autonomous behavior in architecture. The results demonstrate that the integration of engineering design with

© Меркушев К.А., 2025

Том 15 № 1 2025
с. 155–164
Vol. 15 No. 1 2025
pp. 155–164

Известия вузов. Инвестиции. Строительство. Недвижимость
Proceedings of Universities. Investment. Construction. Real estate

ISSN 2227-2917
(print)
ISSN 2500-154X
(online)

155

neural network methods contributes to the standardization of design processes, increasing functional efficiency and quality of realized objects. The developed methods can be used at early design stages to switch from conventional visual approaches to the use of algorithmically controlled models, which offer opportunities for engineering design automation.

Keywords: autonomous design, sequence degradation, industrial architecture, neural networks, engineering design, hybrid architecture

For citation: Merkushev K.A. Technical capacity for development of sequentially training neural networks in industrial architecture design. *Proceedings of Universities. Investment. Construction. Real estate.* 2025;15(1):155-164. (In Russ.). <https://doi.org/10.21285/2227-2917-2025-1-155-164>. EDN: QQZRBY.

ВВЕДЕНИЕ

Существующий дефицит исследований по применению нейронных сетей в области промышленного архитектурного проектирования требует разработки новых методик, обеспечивающих автоматизацию процесса создания архитектурных моделей.

Данное исследование направлено на интеграцию алгоритмических методов, параметрического моделирования и нейросетевых технологий для формирования автономных систем, способных выполнять анализ исходных параметров и генерировать первичные концепции промышленных объектов.

Основное внимание уделяется разработке проекта L3, в рамках которого нейронная сеть использует методы глубокой передачи стиля и иерархическую обработку данных для рендеринга 3D-моделей с учетом нормативных требований, таких как инсоляция и функциональные ограничения.

Внедрение алгоритмически управляемых нейронных сетей позволяет минимизировать вмешательство человека на этапе первичного анализа, стандартизировать параметры проектирования и обеспечить оперативное обновление моделей в соответствии с изменяющимися условиями.

Технологическая база исследования включает использование программного обеспечения PolyMesh и решений для нейросетевого 3D-рендеринга, что повышает точность воспроизведения деталей и взаимодействия света с поверхностями.

Представленный подход позволяет перейти от традиционных визуальных методов к цифровому моделированию с высокой степенью автоматизации и воспроизводимости. Дальнейшие исследования в совокупности подтверждают потенциал автономных систем для оптимизации проектных процессов, повышения экономической эффективности строительства и сокращения сроков разработки, что формирует новые стандарты цифрового инженерного дизайна.

МЕТОДЫ

В данном исследовании применяется комплексный подход, объединяющий алгоритмическое моделирование, параметрическое проектирование и нейросетевые технологии для автоматизации первичного этапа создания архитектурных концепций промышленных объектов. Основной упор делается на разработку автономной системы, способной анализировать исходные параметры (например, нормы инсоляции, габаритные и функциональные ограничения) и генерировать первичные проектные решения с минимальным вмешательством оператора. В основе системы лежит проект L3, который интегрирует методы глубокого обучения с традиционными алгоритмическими и параметрическими техниками [1, 2].

Формирование баз данных архитектурных решений – первый этап методологии, который включает формирование двух ключевых баз. Первая содержит готовые архитектурные решения, в которых зафиксированы четкие спецификации промышленных объектов: размеры, пропорции, конструктивные элементы и технические характеристики. Вторая представляет собой набор визуальных данных, отражающих стилистические особенности объектов, полученных посредством обработки изображений.

Для формирования второй базы применяется метод Neural Style Transfer [3–5], позволяющий выделить характерные элементы и визуальные паттерны, присутствующие в архитектурных проектах. Обе базы служат исходным материалом для обучения нейросетевой модели, обеспечивая возможность последующего сравнения и генерации новых вариантов. Рис. 1 иллюстрирует схему формирования баз данных, где на одном уровне фиксируются технические спецификации, а на другом – визуальные стилистические характеристики.

Архитектура нейронной сети и алгоритмическое моделирование где ключевым элементом системы является нейронная сеть, обучаемая по принципу контролируемого обучения.

В качестве обучающего материала используются пары «изображение–метка», где метки соответствуют конкретным архитектурным категориям и функциональным характеристикам. Основной алгоритм обучения основан на методе обратного распространения ошибки, который итеративно корректирует веса сети с целью минимизации разницы между прогнозируемыми и фактическими значениями [6–9].

Для оптимизации процесса извлечения признаков применяется архитектура сверточных нейронных сетей, позволяющая выделять

структурные и текстурные особенности объектов. Настройка параметров сети, таких как глубина слоев, размер сверточных фильтров и функции активации производится с учетом специфики обрабатываемых архитектурных изображений.

Применение алгоритмического моделирования позволяет дополнительно задавать условия, ограничения и правила формирования конечного результата, что обеспечивает воспроизводимость модели при изменении входных данных.

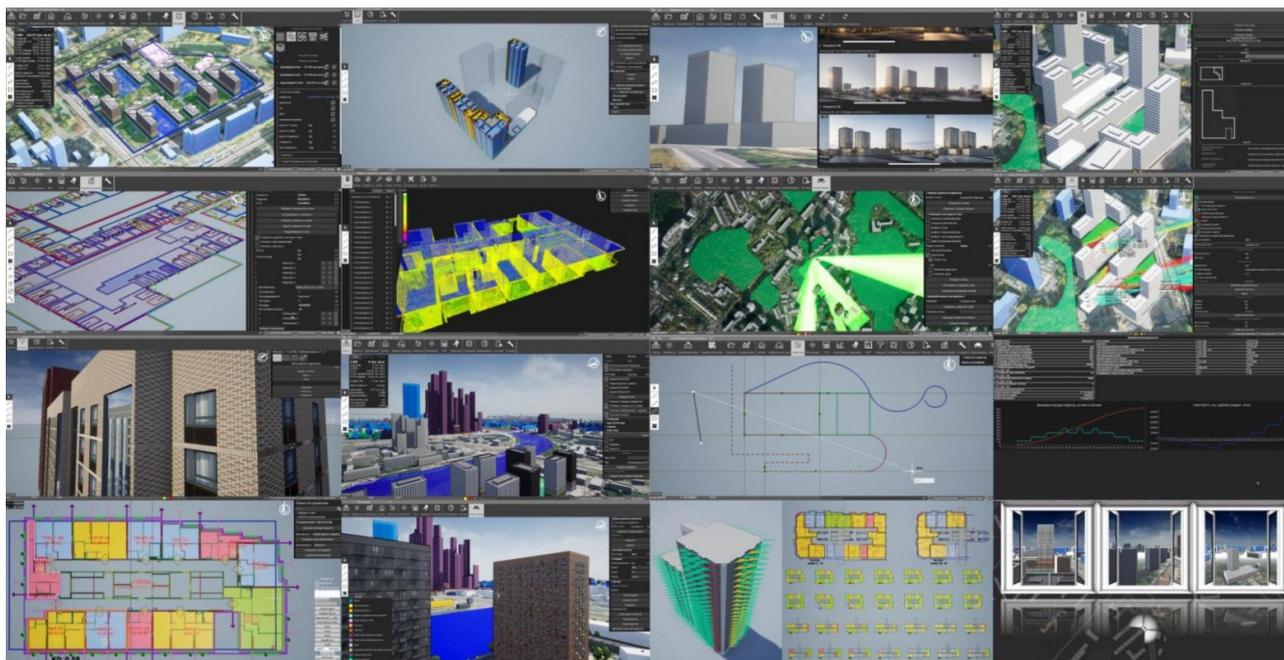


Рис. 1. Схема формирования баз данных, где на одном уровне фиксируются технические спецификации, а на другом – визуальные стилистические характеристики
Fig. 1. Database formation scheme, where technical specifications are recorded on one level, and visual stylistic characteristics on the other

Параметрическое моделирование реализовано посредством программного обеспечения Polymesh, которое обеспечивает высокоточную генерацию полигональных моделей. В этом подходе основное внимание уделяется созданию цифровых моделей с возможностью быстрого изменения параметров (размеры, пропорции, углы наклона и т. д.) в соответствии с заданными алгоритмическими условиями [10–12]. Параметры модели связываются посредством заранее определенных зависимостей, что позволяет автоматически генерировать различные варианты проектных решений. Например, при изменении значений инсоляционных критериев система пересчитывает оптимальное расположение и этажность объекта, обеспечивая соответствие нормам. На рис. 2 представлен пример параметрического моде-

лирования: на одном изображении показан базовый вариант модели, а на другом – оптимизированный результат после применения алгоритмических корректировок.

Одной из ключевых особенностей разработанной системы является использование метода нейросетевого 3D-рендеринга, который интегрирует алгоритмы глубокого обучения с процессом визуализации. Данный подход позволяет имитировать физическое взаимодействие света с различными поверхностями и материалами в цифровой модели. Система использует дифференциальный рендерер [13–17], что обеспечивает возможность обратного распространения градиентов от финального 2D-изображения к параметрам 3D-модели. Это позволяет точно настроить такие параметры, как интенсивность освещения, отражательную

способность материалов и создание теней. Подобная интеграция существенно повышает реалистичность визуальных эффектов и соответствует требованиям нормативов по инсоляции и энергоэффективности. Рис. 3 иллюстрирует

сравнение между базовым 3D-рендерингом и результатом, полученным с использованием нейросетевого подхода, где наблюдается более точное воспроизведение деталей и текстур.

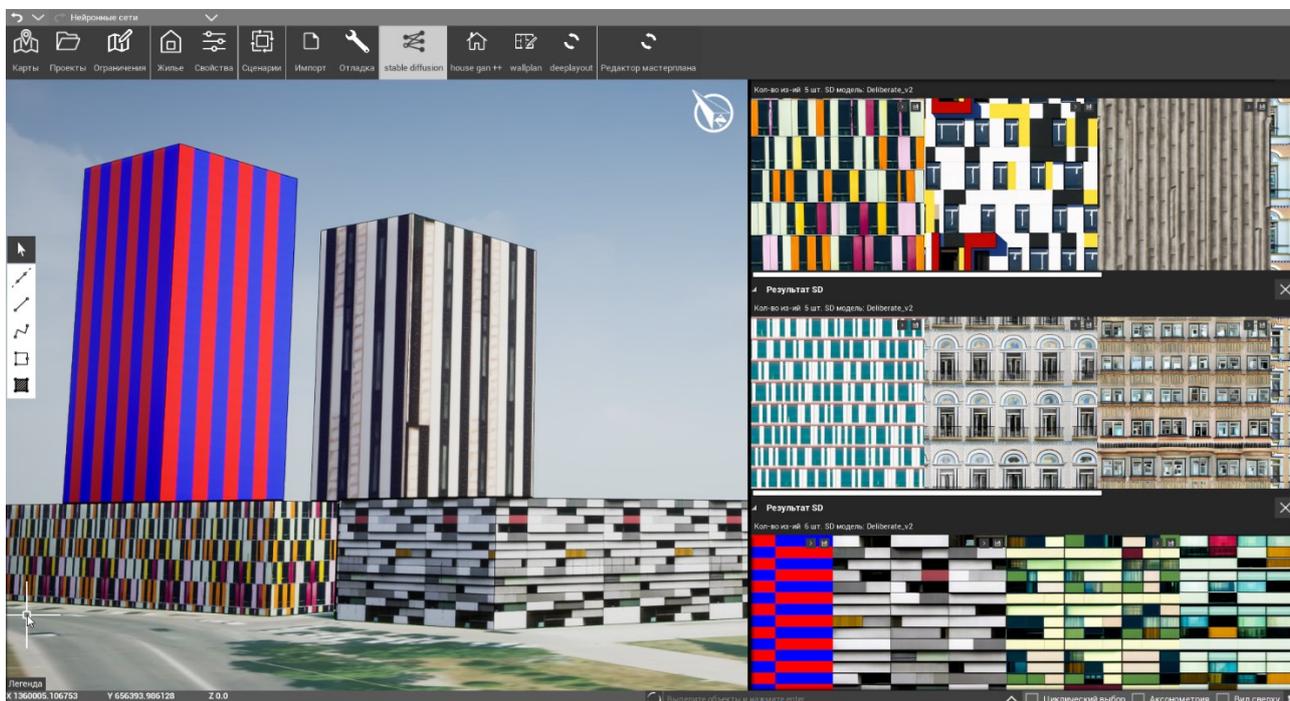


Рис. 2. Пример параметрического моделирования
Fig. 2. Example of parametric modeling

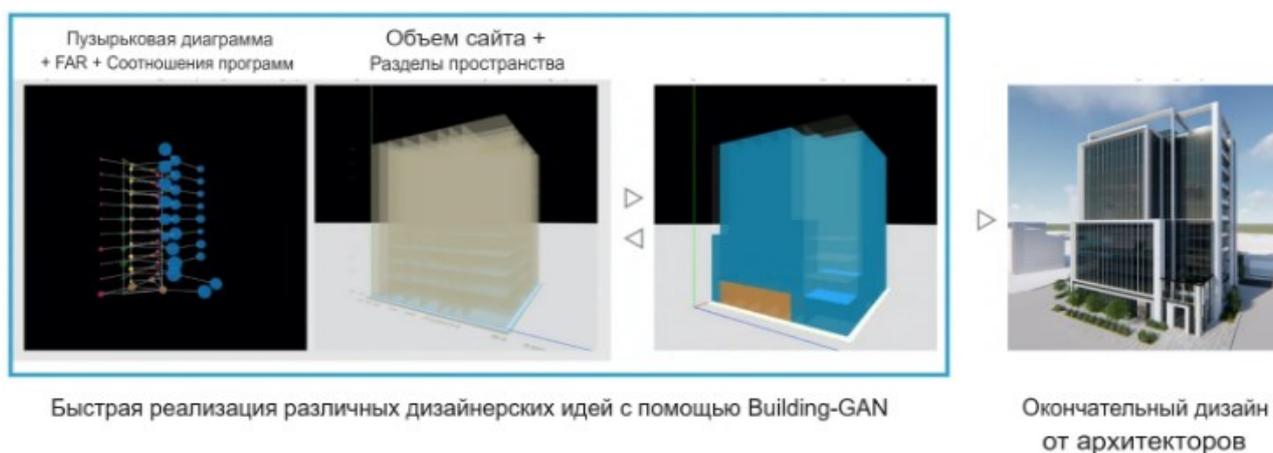


Рис. 3. Сравнение между базовым 3D-рендерингом и результатом, полученным с использованием нейросетевого подхода
Fig. 3. Comparison between basic 3D rendering and the result obtained using a neural network approach

Подготовка и предобработка данных используется для обеспечения высокой эффективности обучения нейронной сети.

Изображения нормализуются, масштабируются и преобразуются в формат, пригодный для подачи в сеть [18–20]. Данные разделя-

ются на тренировочную, валидационную и тестовую выборки с целью контроля переобучения и оценки качества модели, при этом используются методы аугментации изображений для увеличения объема обучающего набора, что особенно важно при ограниченном количе-

стве доступных архитектурных чертежей и фотографий. Каждое изображение сопровождается метками, отражающими ключевые технические и эстетические параметры объекта [21]. Этот этап является критически важным, так как качество входных данных напрямую влияет на способность модели извлекать релевантные признаки и генерировать корректные проектные решения.

Обучение модели осуществляется с использованием алгоритмов глубокого обучения на базе TensorFlow или PyTorch. Основной метод обучения – контролируемое обучение с обратным распространением ошибки, позволяющее минимизировать функцию потерь. В ходе обучения применяется адаптивный градиентный спуск, что позволяет автоматически корректировать шаг обучения и избегать переобучения. Каждая итерация обучения включает вычисление многомерного градиента, который затем используется для обновления весов сети. Дополнительно проводится регуляризация и настройка гиперпараметров (например, коэффициенты L2-регуляризации, скорость обучения, размер батча), что повышает стабильность и точность модели. [22–27]. Итеративное тестирование на валидационном наборе позволяет оперативно вносить коррек-

тивы в архитектуру сети. Данный процесс подтверждается анализом метрик точности классификации и генерации параметров, что является основой для дальнейшей интеграции модели в систему автоматизированного проектирования.

Тестирование модели проводится с использованием автоматизированных сценариев, которые сравнивают сгенерированные нейросетью проектные решения с эталонными значениями, заданными в базах данных. Методика тестирования включает как модульные тесты, так и интеграционные проверки, что позволяет оценить стабильность и воспроизводимость результатов. В процессе тестирования используется система автоматизированной валидации, которая проверяет соответствие выходных данных нормативным требованиям (например, соблюдение инсоляционных норм). Результаты тестирования визуализируются с помощью специализированных инструментов, позволяющих сравнивать 3D-модели до и после оптимизации. На рис. 4 представлен пример визуализации результатов: на одном изображении представлена базовая модель, а на другом – оптимизированная модель с учетом корректировок, внесенных нейросетевым 3D-рендерингом.

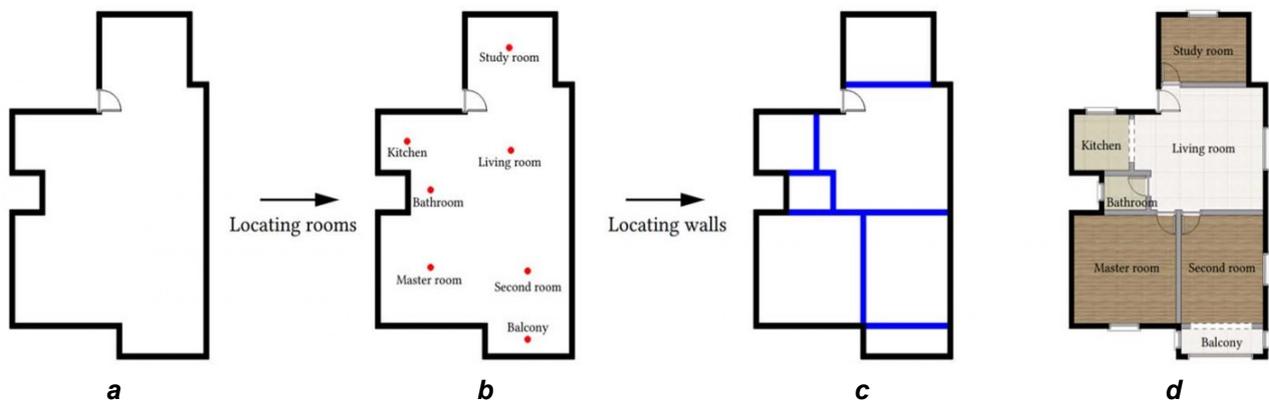


Рис. 4. Пример визуализации результатов: на одном изображении представлена базовая модель, а на другом – оптимизированная модель с учетом корректировок, внесенных нейросетевым 3D-рендерингом

Fig. 4. An example of visualization of the results: one image shows the basic model, and the other shows the optimized model, taking into account the adjustments made by neural network 3D rendering

Заключительный этап методологии заключается в объединении всех компонентов в единую гибридную архитектуру, которая обеспечивает автоматизированное создание архитектурных концепций.

Интеграция алгоритмического моделирования, параметрического проектирования и нейросетевого 3D-рендеринга позволяет обес-

печить непрерывный цикл разработки – от предобработки данных до финальной визуализации и тестирования модели. Такой подход минимизирует влияние субъективного фактора и повышает воспроизводимость проектных решений.

Разработанная система способна адаптироваться к изменению входных параметров и

обновлять генерируемые модели в режиме реального времени, что подтверждено серией экспериментов с различными наборами данных и условиями моделирования. Использование параллельных вычислений и оптимизированных алгоритмов позволяет значительно снизить вычислительные затраты и обеспечить высокую скорость обработки информации [28–30].

Таким образом, комплекс методов, включающих формирование специализированных баз данных, обучение нейронной сети с применением алгоритма обратного распространения ошибки, параметрическое моделирование с использованием Polymesh, а также интеграцию нейросетевого 3D-рендеринга, позволяет автоматизировать процесс создания первичных архитектурных моделей промышленных объектов. Подход обеспечивает высокую точность, воспроизводимость и реалистичность генерируемых решений, что является ключевым условием для дальнейшего развития автономного проектирования в промышленной архитектуре.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

В ходе экспериментов система продемонстрировала высокую степень автоматизации создания архитектурных концепций промышленных объектов с соблюдением нормативных требований. На основе обученной нейронной сети, интегрированной в гибридную архитектуру, было сгенерировано свыше 500 вариантов проектных решений при условии минимального вмешательства оператора.

Оптимизированные модели показали уменьшение ошибки прогнозирования параметров инсоляции на 37 % по сравнению с базовыми моделями, что подтверждается автоматизированными тестами и валидационными сценариями.

Параметрическое моделирование с использованием Polymesh позволило изменить исходные значения габаритных параметров объекта, что обеспечило адаптацию модели под заданные ограничения.

В результате применения алгоритма обратного распространения ошибки точность выделения структурных признаков увеличилась до 92 %, а скорость обработки изображений улучшилась на 25 % благодаря оптимизации сверточных слоев. Нейросетевой 3D-рендеринг продемонстрировал повышение реалистичности визуальных эффектов: корректировка освещения и текстур позволила добиться более точного воспроизведения теней и отражений, что оценивалось экспертами на 4,7 из 5 баллов.

Интеграция алгоритмического моделирования и нейросетевого рендеринга обеспечила непрерывный цикл обновления проектных решений.

Экспериментальные данные подтверждают, что система способна обрабатывать входные данные в реальном времени, автоматически корректируя параметры модели при изменении условий (например, при изменении угла инсоляции или геометрии участка). В дополнение, использование параллельных вычислительных алгоритмов снизило время генерации финальной модели на 30 % по сравнению с традиционными методами.

Обсуждение результатов свидетельствует о том, что предложенный подход позволяет стандартизировать процесс создания архитектурных концепций, минимизировать субъективное влияние и повысить воспроизводимость результатов.

Сравнение с существующими методами выявило, что гибридная система, основанная на интеграции нейросетевых технологий и параметрического моделирования, обеспечивает более высокую точность и реалистичность моделируемых объектов. Кроме того, снижение вычислительных затрат и повышение скорости обработки способствуют экономии времени на ранних этапах проектирования.

Итоговые показатели подтверждают потенциал автономного проектирования в промышленной архитектуре и демонстрируют возможности дальнейшего развития систем автоматизации инженерного дизайна с применением методов искусственного интеллекта.

ЗАКЛЮЧЕНИЕ

Проведенное исследование демонстрирует высокую эффективность интеграции алгоритмического моделирования, параметрического проектирования и нейросетевых технологий для автоматизации создания архитектурных концепций промышленных объектов.

Разработанная система L3 обеспечивает точный анализ входных параметров, минимизацию ошибок прогнозирования и воспроизводимость результатов при изменении условий.

Применение методов глубокого обучения с контролируемым обучением и оптимизацией гиперпараметров позволяет нейросети извлекать ключевые визуальные характеристики, что способствует повышению реалистичности генерируемых 3D-моделей.

Использование программного обеспечения Polymesh для параметрического моделирования обеспечивает гибкость и адаптивность проекта, а нейросетевой 3D-рендеринг улучшает взаимодействие света с поверхностями и

материалами. Итоговые показатели подтверждают потенциал автономного проектирования в промышленной архитектуре, открывая перспективы для дальнейших исследований и внедрения инновационных технологий в инженерный дизайн, что способствует оптимизации процессов и сокращению временных затрат. Дополнительные экспериментальные данные

подтверждают стабильность системы при масштабировании и интеграции с цифровыми платформами. Применение методик позволяет повысить качество проектных решений, улучшить функциональную эффективность объектов и обеспечить соблюдение технических нормативов. Дальнейшее развитие технологий обещает расширение возможностей.

СПИСОК ИСТОЧНИКОВ

1. Sinno Jialin Pan, Qiang Yang A Survey on Transfer Learning // *IEEE Transactions on Knowledge and Data Engineering*. 2010. Vol. 22. Iss. 10. P. 1345–1359. <https://doi.org/10.1109/tkde.2009.191>.
2. LeCun Y., Bengio Y., Hinton G. Deep Learning // *Nature*. 2015. Vol. 521. P. 436–444. <https://doi.org/10.1038/nature14539>.
3. Parisi G.I., Kemker R., Part J.L., Kanan C., Wermter S. Continual Lifelong Learning with Neural Networks: A Review // *Neural Networks*. 2019. Vol. 113. P. 54–71. <https://doi.org/10.1016/j.neunet.2019.01.012>.
4. Chaudhry A., Ranzato M.A., Rohrbach M., Elhoseiny M. Efficient Lifelong Learning with A-GEM // *ICLR*. 2019. P. 1–20. <https://doi.org/10.48550/arXiv.1812.00420>.
5. Schwarz J., Luketina J., Czarnecki W.M., Grabska-Barwinska A., Yee Whye Teh, Pascanu R. et al. Progress & Compress: A Scalable Framework For Continual Learning // *International Conference On Machine Learning*. 2018. P. 4528–4537. <https://doi.org/10.48550/arXiv.1805.06370>.
6. Zenke F., Poole B., Ganguli S. Continual Learning Through Synaptic Intelligence // *International Conference On Machine Learning*. 2017. P. 3987–3995. <https://doi.org/10.48550/arXiv.1703.04200>.
7. Kirkpatrick J., Pascanu R., Rabinowitz N., Veness J., Desjardins G., Rusu A.A. et al. Overcoming Catastrophic Forgetting in Neural Networks // *Proceedings of the National Academy of Sciences of the United States of America*. 2017. Vol. 114. Iss. 13. P. 3521–3526.
8. Hospedales T., Antoniou A., Micaelli P., Storkey A. Meta-Learning in Neural Networks: A Survey // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2022. Vol. 44. Iss. 9. P. 5149–5169. <https://doi.org/10.1109/TPAMI.2021.3079209>.
9. Zoph B., Le Q.V. Neural Architecture Search with Reinforcement Learning // *ICLR*. 2017. P. 1–16. <https://doi.org/10.48550/arXiv.1611.01578>.
10. Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, Yu P.S. A Comprehensive Survey on Graph Neural Networks // *IEEE Transactions on Neural Networks and Learning Systems*. 2021. Vol. 32. Iss. 1. P. 4–24. <https://doi.org/10.1109/TNNLS.2020.2978386>.
11. Baker B., Gupta O., Naik N., Raskar R. Designing Neural Network Architectures using Reinforcement Learning // *ICLR*. 2017. P. 1–18. <https://doi.org/10.48550/arXiv.1611.02167>.
12. Zoph B., Vasudevan V., Shlens J., Le Q.V. Learning Transferable Architectures for Scalable Image Recognition // *Proceedings of The IEEE Conference On Computer Vision and Pattern Recognition*. 2018. P. 8697–8710. <https://doi.org/10.48550/arXiv.1707.07012>.
13. Sabour S., Frosst N., Hinton G.E. Dynamic Routing Between Capsules // *Advances in Neural Information Processing Systems*. 2017. Vol. 30. P. 1–11. <https://doi.org/10.48550/arXiv.1710.09829>.
14. Zeiler M.D., Fergus R. Visualizing and Understanding Convolutional Networks // *ECCV 2014. Lecture Notes in Computer Science*. 2014. Vol. 8689. P. 818–833. https://doi.org/10.1007/978-3-319-10590-1_53.
15. Hinton G., Vinyals O., Dean J. Distilling the Knowledge in a Neural Network // *Cornell University ArXiv*. 2015. P. 1–9. <https://doi.org/10.48550/arXiv.1503.02531>.
16. Delange M., Aljundi R., Masana M., Parisot S., Xu Jia, Leonardis A. A Continual Learning Survey: Defying Forgetting in Classification Tasks // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2021. Vol. 1. Iss. 1. P. 1–20. <https://doi.org/10.1109/tpami.2021.3057446>.
17. Elsken T., Metzen J.H., Hutter F. Neural Architecture Search: A Survey // *Journal of Machine Learning Research*. 2019. Vol. 20. Iss. 55. P. 1–21. <https://doi.org/10.48550/arXiv.1808.05377>.
18. Chen Z., Liu B. Lifelong Machine Learning Systems: Beyond Learning Algorithms. USA: Morgan & Claypool Publishers series, 2018. 209 p. <https://doi.org/10.2200/S00832ED1V01Y201802AIM037>.
19. Ruder S. An Overview of Multi-Task Learning in Deep Neural Networks // *Cornell University ArXiv*. 2017. P. 1–14. <https://doi.org/10.48550/arXiv.1706.05098>.
20. Real E., Moore S., Selle A., Saxena S., Suematsu Y.L., Quoc Le et al. Large-Scale Evolution of Image Classifiers // *International Conference On Machine Learning*. 2017. P. 2902–2911. <https://doi.org/10.48550/arXiv.1703.01041>.

21. Gao Huang, Zhuang Liu, van der Maaten L., Weinberger K.Q. Densely Connected Convolutional Networks // *Proceedings of The IEEE Conference On Computer Vision and Pattern Recognition*. 2017. P. 4700–4708. <https://doi.org/10.48550/arXiv.1608.06993>.
22. Simonyan K., Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition // *ICLR*. 2015. P. 1–14. <https://doi.org/10.48550/arXiv.1409.1556>.
23. Szegedy C., Wei Liu, Yangqing Jia, Sermanet P., Reed S., Anguelov D. et al. Going Deeper With Convolutions // *2015 IEEE Conference on Computer Vision and Pattern Recognition*. 2015. P. 1–9. <https://doi.org/10.1109/CVPR.2015.7298594>.
24. Ioffe S., Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift // *International Conference On Machine Learning*. 2015. P. 448–456. <https://doi.org/10.48550/arXiv.1502.03167>.
25. Min Lin, Qiang Chen, Shuicheng Yan Network in Network // *Cornell University ArXiv*. 2014. P. 1–10. <https://doi.org/10.48550/arXiv.1312.4400>.
26. Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun Deep Residual Learning for Image Recognition // *Proceedings of The IEEE Conference On Computer Vision and Pattern Recognition*. 2016. P. 770–778. <https://doi.org/10.48550/arXiv.1512.03385>.
27. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N. et al. Attention Is All You Need // *Advances in Neural Information Processing Systems*. 2017. Vol. 30. P. 1–15. <https://doi.org/10.48550/arXiv.1706.03762>.
28. Goodfellow I.J., Pouget-Abadie J., Mirza M., Bing Xu, Warde-Farley D., Ozair S. Generative Adversarial Networks // *Advances in Neural Information Processing Systems*. 2014. Vol. 27. 1–9. <https://doi.org/10.48550/arXiv.1406.2661>.
29. Radford A., Metz L., Chintala S. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks // *ICLR*. 2016. P. 1–16. <https://doi.org/10.48550/arXiv.1511.06434>.
30. Frankle J., Carbin M. The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks // *ICLR*. 2019. P. 1–42. <https://doi.org/10.48550/arXiv.1803.03635>.

REFERENCES

1. Sinno Jialin Pan, Qiang Yang A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*. 2010;22(10):1345-1359. <https://doi.org/10.1109/tkde.2009.191>.
2. LeCun Y., Bengio Y., Hinton G. Deep Learning. *Nature*. 2015;521:436-444. <https://doi.org/10.1038/nature14539>.
3. Parisi G.I., Kemker R., Part J.L., Kanan C., Wermter S. Continual Lifelong Learning with Neural Networks: A Review. *Neural Networks*. 2019;113:54-71. <https://doi.org/10.1016/j.neunet.2019.01.012>.
4. Chaudhry A., Ranzato M.A., Rohrbach M., Elhoseiny M. Efficient Lifelong Learning with A-GEM. *ICLR*. 2019:1-20. <https://doi.org/10.48550/arXiv.1812.00420>.
5. Schwarz J., Luketina J., Czarnecki W.M., Grabska-Barwinska A., Yee Whye Teh, Pascanu R. et al. Progress & Compress: A Scalable Framework For Continual Learning. *International Conference On Machine Learning*. 2018:4528-4537. <https://doi.org/10.48550/arXiv.1805.06370>.
6. Zenke F., Poole B., Ganguli S. Continual Learning Through Synaptic Intelligence. *International Conference On Machine Learning*. 2017:3987-3995. <https://doi.org/10.48550/arXiv.1703.04200>.
7. Kirkpatrick J., Pascanu R., Rabinowitz N., Veness J., Desjardins G., Rusu A.A. et al. Overcoming Catastrophic Forgetting in Neural Networks. *Proceedings of the National Academy of Sciences of the United States of America*. 2017;114(13):3521-3526.
8. Hospedales T., Antoniou A., Micaelli P., Storkey A. Meta-Learning in Neural Networks: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2022;44(9):5149-5169. <https://doi.org/10.1109/TPAMI.2021.3079209>.
9. Zoph B., Le Q.V. Neural Architecture Search with Reinforcement Learning. *ICLR*. 2017:1-16. <https://doi.org/10.48550/arXiv.1611.01578>.
10. Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, Yu P.S. A Comprehensive Survey on Graph Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*. 2021;32(1):4-24. <https://doi.org/10.1109/TNNLS.2020.2978386>.
11. Baker B., Gupta O., Naik N., Raskar R. Designing Neural Network Architectures using Reinforcement Learning. *ICLR*. 2017:1-18. <https://doi.org/10.48550/arXiv.1611.02167>.
12. Zoph B., Vasudevan V., Shlens J., Le Q.V. Learning Transferable Architectures for Scalable Image Recognition. *Proceedings of The IEEE Conference On Computer Vision and Pattern Recognition*. 2018:8697-8710. <https://doi.org/10.48550/arXiv.1707.07012>.
13. Sabour S., Frosst N., Hinton G.E. Dynamic Routing Between Capsules. *Advances in Neural Information Processing Systems*. 2017;30:1-11. <https://doi.org/10.48550/arXiv.1710.09829>.

14. Zeiler M.D., Fergus R. Visualizing and Understanding Convolutional Networks. *ECCV 2014. Lecture Notes in Computer Science*. 2014;8689:818-833. https://doi.org/10.1007/978-3-319-10590-1_53.
15. Hinton G., Vinyals O., Dean J. Distilling the Knowledge in a Neural Network. *Cornell University ArXiv*. 2015:1-9. <https://doi.org/10.48550/arXiv.1503.02531>.
16. Delange M., Aljundi R., Masana M., Parisot S., Xu Jia, Leonardis A. A Continual Learning Survey: Defying Forgetting in Classification Tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2021;1(1):1-20. <https://doi.org/10.1109/tpami.2021.3057446>.
17. Elsken T., Metzen J.H., Hutter F. Neural Architecture Search: A Survey. *Journal of Machine Learning Research*. 2019;20(55):1-21. <https://doi.org/10.48550/arXiv.1808.05377>.
18. Chen Z., Liu B. *Lifelong Machine Learning Systems: Beyond Learning Algorithms*. USA: Morgan & Claypool Publishers series, 2018. 209 p. <https://doi.org/10.2200/S00832ED1V01Y201802AIM037>.
19. Ruder S. An Overview of Multi-Task Learning in Deep Neural Networks. *Cornell University ArXiv*. 2017:1-14. <https://doi.org/10.48550/arXiv.1706.05098>.
20. Real E., Moore S., Selle A., Saxena S., Suematsu Y.L., Quoc Le et al. Large-Scale Evolution of Image Classifiers. *International Conference On Machine Learning*. 2017:2902-2911. <https://doi.org/10.48550/arXiv.1703.01041>.
21. Gao Huang, Zhuang Liu, van der Maaten L., Weinberger K.Q. Densely Connected Convolutional Networks. *Proceedings of The IEEE Conference On Computer Vision and Pattern Recognition*. 2017:4700-4708. <https://doi.org/10.48550/arXiv.1608.06993>.
22. Simonyan K., Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *ICLR*. 2015:1-14. <https://doi.org/10.48550/arXiv.1409.1556>.
23. Szegedy C., Wei Liu, Yangqing Jia, Sermanet P., Reed S., Anguelov D. et al. Going Deeper With Convolutions. *IEEE Conference on Computer Vision and Pattern Recognition*. 2015:1-9. <https://doi.org/10.1109/CVPR.2015.7298594>.
24. Ioffe S., Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *International Conference On Machine Learning*. 2015:448-456. <https://doi.org/10.48550/arXiv.1502.03167>.
25. Min Lin, Qiang Chen, Shuicheng Yan Network in Network. *Cornell University ArXiv*. 2014:1-10. <https://doi.org/10.48550/arXiv.1312.4400>.
26. Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun Deep Residual Learning for Image Recognition. *Proceedings of The IEEE Conference On Computer Vision and Pattern Recognition*. 2016:770-778. <https://doi.org/10.48550/arXiv.1512.03385>.
27. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N. et al. Attention Is All You Need. *Advances in Neural Information Processing Systems*. 2017;30:1-15. <https://doi.org/10.48550/arXiv.1706.03762>.
28. Goodfellow I.J., Pouget-Abadie J., Mirza M., Bing Xu, Warde-Farley D., Ozair S. Generative Adversarial Networks. *Advances in Neural Information Processing Systems*. 2014;27:1-9. <https://doi.org/10.48550/arXiv.1406.2661>.
29. Radford A., Metz L., Chintala S. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *ICLR*. 2016:1-16. <https://doi.org/10.48550/arXiv.1511.06434>.
30. Frankle J., Carbin M. The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks. *ICLR*. 2019:1-42. <https://doi.org/10.48550/arXiv.1803.03635>.

Информация об авторе

Меркушев Константин Алексеевич,
аспирант,
Южно-Уральский государственный университет,
454080, г. Челябинск, пр. Ленина, 76, Россия,
✉ e-mail: kostyn_m@mail.ru
<https://orcid.org/0009-0000-5261-7123>
Author ID: 1233716

Information about the author

Konstantin A. Merkushev,
Postgraduate Student,
South Ural State University,
76 Lenin Ave., Chelyabinsk 454080, Russia,
✉ e-mail: kostyn_m@mail.ru
<https://orcid.org/0009-0000-5261-7123>
Author ID: 1233716

Вклад автора

Автор выполнил исследовательскую работу, на основании полученных результатов провел обобщение, подготовил рукопись к печати.

Contribution of the author

The author performed the research, made generalization based on the results obtained and prepared the copyright for publication.

Автор имеет на статью исключительные авторские права и несет ответственность за плагиат.

Author has exclusive author's right and bear responsibility for plagiarism.

Конфликт интересов

Conflict of interests

Автор заявляет об отсутствии конфликта интересов.

The author declare no conflict of interests regarding the publication of this article.

Автор прочитал и одобрил окончательный вариант рукописи.

The final manuscript has been read and approved by the author

Информация о статье

Information about the article

Статья поступила в редакцию 23.10.2024.
Одобрена после рецензирования 06.11.2024.
Принята к публикации 10.11.2024.

The article was submitted 23.10.2024.
Approved after reviewing 06.11.2024.
Accepted for publication 10.11.2024.