

УДК 004.8

DOI: [10.26102/2310-6018/2025.51.4.025](https://doi.org/10.26102/2310-6018/2025.51.4.025)

Применение глубокого обучения с подходом «один против всех» для задачи многоклассовой классификации дефектов металлических поверхностей

В.Е. Сосновская, А.Д. Гаврилова, Е.А. Волкова, И.Д. Котилевец, Д.Ю. Ильин✉

МИРЭА – Российский технологический университет, Москва, Российская Федерация

Резюме. В статье исследуется задача многоклассовой классификации дефектов металлических поверхностей с использованием методов глубокого обучения. В качестве основного подхода применяется стратегия «один против всех», позволяющая эффективно разделять различные классы дефектов. Для первоначального анализа использовался датасет «NEU», который включал в себя 6 классов повреждений. Полученные оценки качества классификации сравнивались с другими существующими решениями, после чего набор данных был дополнен одним классом из датасета «Severstal: Steel Defect Detection». Предложены две архитектуры сверточных нейронных сетей, каждая из которых была адаптирована под соответствующие классы. Первая архитектура включает в себя 5 сверточных слоев, 5 слоев максимизирующей подвыборки и 2 полносвязных слоя. Вторая содержит на 2 слоя больше: дополнительный сверточный слой и слой максимизирующей подвыборки. Тестирование на данных датасета «NEU» показало высокую эффективность: доля правильных ответов итоговой модели достигает 98,33 %, точность – 98,39 %, полнота – 98,33 % и F1-мера – 98,33 %. Анализ результатов показал, что предложенный подход продемонстрировал эффективность, сравнимую с результатами из других исследований, а предложенные архитектуры не уступают современным решениям. Модель демонстрирует хорошую скорость обработки, до 103 кадров в секунду на центральном процессоре, что делает ее применимой в производстве, позволяя обнаруживать повреждения в реальном времени. После добавления дополнительного класса доля правильных ответов стала равна 97,14 %, точность – 97,24 %, полнота – 97,14 % и F1-мера – 97,12 %, что указывает на устойчивость и масштабируемость предложенного решения на основе подхода «one-vs-all».

Ключевые слова: нейронные сети, сверточные нейронные сети, датасет, классификация, дефекты металлических поверхностей, глубокое обучение.

Для цитирования: Сосновская В.Е., Гаврилова А.Д., Волкова Е.А., Котилевец И.Д., Ильин Д.Ю. Применение глубокого обучения с подходом «один против всех» для задачи многоклассовой классификации дефектов металлических поверхностей *Моделирование, оптимизация и информационные технологии*. 2025;13(4). URL: <https://moitvvt.ru/ru/journal/pdf?id=2032> DOI: 10.26102/2310-6018/2025.51.4.025

Application of deep learning with the one-vs-all approach for the task of multiclass classification of metal surface defects

V.E. Sosnovskaya, A.D. Gavrilova, E.A. Volkova, I.D. Kotilevets, D.Y. Ilin✉

MIREA – Russian Technological University, Moscow, the Russian Federation

Abstract. The article investigates the problem of multiclass classification of metal surface defects using deep learning methods. The primary approach employs a "one-vs-all" strategy, which effectively separates different defect classes. Initial analysis utilized the NEU dataset, comprising six defect categories. The resulting metrics were compared against existing solutions, after which the dataset was extended with an additional class from the "Severstal: Steel Defect Detection" dataset. Two convolutional neural network architectures were proposed, each tailored to the respective set of classes. The first architecture consists of five convolutional layers, five max-pooling layers, and two fully

connected layers. The second architecture includes two additional layers: an extra convolutional layer and an additional max-pooling layer. Evaluation on the NEU dataset demonstrated high performance: the final model achieved an accuracy of 98.33 %, precision of 98.39 %, recall of 98.33 %, and an F1-score of 98.33 %. Analysis of the results showed that the proposed approach achieves performance comparable to other research results, and the proposed architectures are on par with state-of-the-art solutions. The model also exhibits good processing speed – up to 103 frames per second on the CPU – making it suitable for industrial deployment and enabling real-time defect detection. After extending the solution with the additional class, the model maintained strong performance, achieving an accuracy of 97.14 %, precision of 97.24 %, recall of 97.14 %, and an F1-score of 97.12 %, which suggests robustness and scalability of the proposed solution based on the "one-vs-all" approach.

Keywords: neural networks, convolutional neural networks, dataset, classification, defects of metal surfaces, deep learning.

For citation: Sosnovskaya V.E., Gavrilova A.D., Volkova E.A., Kotilevets I.D., Ilin D.Y. Application of deep learning with the one-vs-all approach for the task of multiclass classification of metal surface defects. *Modeling, Optimization and Information Technology*. 2025;13(4). (In Russ.). URL: <https://moitvvt.ru/ru/journal/pdf?id=2032> DOI: 10.26102/2310-6018/2025.51.4.025

Введение

Автоматизация процессов визуального контроля металлических изделий становится неотъемлемой частью современной промышленности. Раннее выявление дефектов металлических поверхностей позволяет не только сократить процент брака, но и повысить безопасность эксплуатации конечной продукции.

Подробно перечень дефектов представлен в ГОСТ 21014-2022 «Металлопродукция из стали и сплавов. Дефекты поверхности. Термины и определения»¹. В исследованиях же как правило используются готовые наборы данных, имеющие отличающиеся наименования. Одним из наиболее распространенных наборов данных для обучения и тестирования моделей является датасет «NEU Surface Defect Database». Данный набор включает шесть классов дефектов: «Crazing», «Inclusion», «Patches», «Pitted Surface», «Rolled-in scale» и «Scratches». Датасет широко используется для тестирования различных архитектур нейронных сетей, включая VGG [1], InceptionV3 [2] и SqueezeNet [3]. Благодаря относительно небольшому объему данных, он часто применяется для оценки моделей в условиях ограниченной выборки [4]. Помимо основной версии датасета NEU, существует версия NEU-DET для детекции, включающая в себя аннотацию с координатами повреждений для каждого изображения, однако и эту версию используют для задачи классификации дефектов [5]. Другим популярным датасетом является «Severstal: Steel Defect Detection», применяемый в соревнованиях и научных работах по автоматическому распознаванию дефектов. Он содержит четыре класса, изначально обозначенные условно как «Class1», «Class2», «Class3» и «Class4», без явного описания типов повреждений. Как показано в исследованиях, для определения соответствия классов конкретным дефектам требуется визуальный анализ и сопоставление с другими датасетами [6].

Одним из эффективных методов решения задачи классификации дефектов металлических поверхностей является применение методов глубокого обучения [7, 8], в частности – сверточных нейронных сетей (CNN) [9]. Как правило, наиболее высокие результаты достигаются за счет модификаций современных архитектур глубокого обучения. В частности, усовершенствованные версии ResNet, представленные в работах

¹ ГОСТ 21014-2022. Металлопродукция из стали и сплавов. Дефекты поверхности. Термины и определения: межгосударственный стандарт: издание официальное: введен в действие Приказом Федерального агентства по техническому регулированию и метрологии от 31 мар-та 2022 г. № 182-ст: введен взамен ГОСТ 21014-88: дата введения 2022-09-01. Москва: Российский институт стандартизации; 2022. 97 с.

[6, 10], а также в [11], обеспечивают стабильно высокую точность классификации, от 95,81 % до 97,50 %. Также эффективны модификации семейства Inception [2], показавшие 92–98 % точности. Помимо этого, DenseNet и его расширенные варианты показывают выдающиеся результаты [12, 13], достигая 96–99 % точности. Помимо классических CNN, активно исследуются и альтернативные архитектуры, например, LMViT, который превзошел многие трансформерные модели, достигнув 97,22 % точности [5]. Для задач с ограниченными вычислительными ресурсами также предлагаются компактные решения, например, LCNN [14] с точностью 91,67 % или MobileNetV2 [4, 13], итоговая точность которой варьируется от 87 до 98 %. Таким образом, современные подходы к классификации дефектов на металлических поверхностях охватывают как «тяжелые», высокоточные архитектуры, так и эффективные «легкие» модели, обеспечивая высокую точность в зависимости от требований к скорости обработки данных, ресурсам и надежности.

В существующих исследованиях, как правило, рассматриваются ограниченные наборы типов повреждений, в большинстве проанализированных работ исследуется не более 6 классов дефектов. При этом в стандарте ГОСТ количество возможных повреждений значительно больше. В связи с этим целесообразным становится рассмотрение подхода «один против всех», который позволяет гибко расширять перечень классов, в том числе без полного переобучения всех моделей: для добавления нового класса может быть достаточно обучения лишь одного дополнительного бинарного классификатора.

К тому же, при использовании многоклассовой классификации на основе единой модели могут возникать трудности, связанные с путаницей между отдельными классами [3]. Такое поведение может быть обусловлено схожестью визуальных характеристик, так как дефекты одного класса могут быть очень похожи на дефекты другого [11], или сложностью выделения уникальных признаков для определенных категорий внутри общей модели. Отдельные классы могут иметь специфические особенности, такие как малая выраженность признаков или неоднородность формы. В связи с этим использование стратегии «один против всех» позволяет повысить точность за счет индивидуального обучения моделей для каждого класса, учитывая его уникальные характеристики, и избежать конкуренции за внимание внутри единой модели.

Цель настоящей работы – рассмотреть применение подхода «один против всех» для решения задачи многоклассовой классификации дефектов металлических поверхностей с использованием сверточных нейронных сетей. Таким образом, настоящая работа относится к области компьютерного зрения, в частности, к применению сверточных нейронных сетей для многоклассовой классификации изображений.

Дальнейшая структура статьи включает следующие разделы. В разделе «Материалы и методы» приведено описание используемых архитектур нейронных сетей, включая структуру и параметры слоев, а также подход к подготовке данных и обучению модели. Описаны метрики, которые используются для оценки качества работы модели. В разделе «Результаты» представлены полученные оценки качества и рассчитано время, требуемое для классификации. Проведен более детальный анализ результатов, касающийся каждого типа повреждений. В разделе «Обсуждение» рассматриваются существующие решения и сравниваются результаты, выделяются преимущества выбранного подхода. Кроме того, рассматриваются возможные направления дальнейших исследований и способы улучшения полученных результатов. «Закключение» содержит основные выводы работы.

Материалы и методы

Исходные данные и предварительная обработка. Данные для обучения были взяты из нескольких источников. Первый: «Датасет дефектов стали Северо-Восточного университета (NEU, г. Бостон, США)»², из него было взято шесть классов дефектов («Crazing», «Inclusion», «Patches», «Pitted Surface», «Rolled-in scale» и «Scratches»). Набор включал в себя 1800 изображений (по 300 для каждого класса) с разрешением 200×200 пикселей. Изображения каждого класса были разделены на обучающую, тестовую и проверочную выборки, где 80 процентов изображений использовались для обучения, 10 для проверки и 10 для тестирования. Таким образом, обучающий набор содержал 240 изображений, а тренировочный и тестовый по 30. Целью обучения было добиться высоких результатов на данном датасете, после чего добавить новый класс из другого датасета для демонстрации работы подхода «один против всех» и возможности дообучения системы классификации без необходимости в переобучении моделей.

Второй датасет: «Северсталь: обнаружение дефектов стали»³, из которого был взят класс «Class4», в котором визуально было выявлено повреждение, идентичное расслоению, в связи с чем далее в работе он будет именоваться «Peeling». Набор изображений этого класса включал 801 фотографию с разрешением 1600×256 пикселей. Так как количество примеров для класса «Peeling» было больше, чем для классов в предыдущем датасете, было выполнено сокращение числа изображений (under-sampling), чтобы избежать смещения моделей в сторону класса с большим количеством данных. Было выбрано 300 изображений, которые были разделены на три выборки в соотношении 80–10–10, аналогично прошлому набору данных. Размер изображений также являлся неподходящим для обучения, поэтому все изображения были вручную обрезаны до квадратов 256×256 в том месте, где находилось повреждение, после чего в процессе обучения приводились к разрешению 200×200, что совпадало с размером изображений в прошлом датасете.

Для предотвращения проблем нехватки данных и чувствительности моделей к яркости изображений была проведена аугментация всех изображений, относящихся к тренировочной выборке. Для каждого изображения было добавлено две новых аугментированных версии: с увеличением и уменьшением яркости в полтора раза. Таким образом, количество тренировочных изображений в датасете было увеличено с 240 до 720.

Для предварительной обработки изображений и дополнительных аугментаций в процессе обучения применялся класс ImageDataGenerator из библиотеки Keras, благодаря его способности интегрироваться непосредственно в генератор данных модели. Для тренировочных данных использовалась нормализация изображений, с помощью которой все пиксели были преобразованы из диапазона от 0 до 255 в диапазон от 0 до 1. Это ускоряет обучение нейронной сети, так как градиентный спуск эффективнее работает с малыми значениями, при этом большие значения могут привести к высоким градиентам и нестабильному обучению. Для данных обучающей выборки применялись некоторые типы аугментаций в процессе обучения, такие как случайное вращение изображений на ±20 градусов и отражение изображения по горизонтали с вероятностью 50 процентов. После вращения пустые пиксели заполнялись значениями ближайших известных пикселей. Для тестовых и проверочных данных использовалась только нормализация.

² Md Fantacher Islam. Metal Surface Defects Dataset. Kaggle. URL: <https://www.kaggle.com/datasets/fantacher/neu-metal-surface-defects-data/data> (дата обращения: 28.02.2025).

³ Severstal. Severstal: Steel Defect Detection. Kaggle. URL: <https://www.kaggle.com/c/severstal-steel-defect-detection> (дата обращения: 28.02.2025).

Модель глубокого обучения для многоклассовой классификации. Для решения поставленной задачи была выбрана стратегия «один против всех» с подготовкой нескольких моделей глубокого обучения, каждая из которых должна соответствовать классу дефекта и определять, принадлежит изображение этому классу или нет.

Для обучения использовалась последовательная модель (Sequential) из библиотеки Keras. Это контейнер для слоев, в котором данные проходят через каждый слой по порядку. Архитектуры моделей включают в себя сверточные слои, слои подвыборки, полносвязные слои и регуляризационные механизмы. Две архитектуры моделей и соответствующие им классы показаны на Рисунке 1.

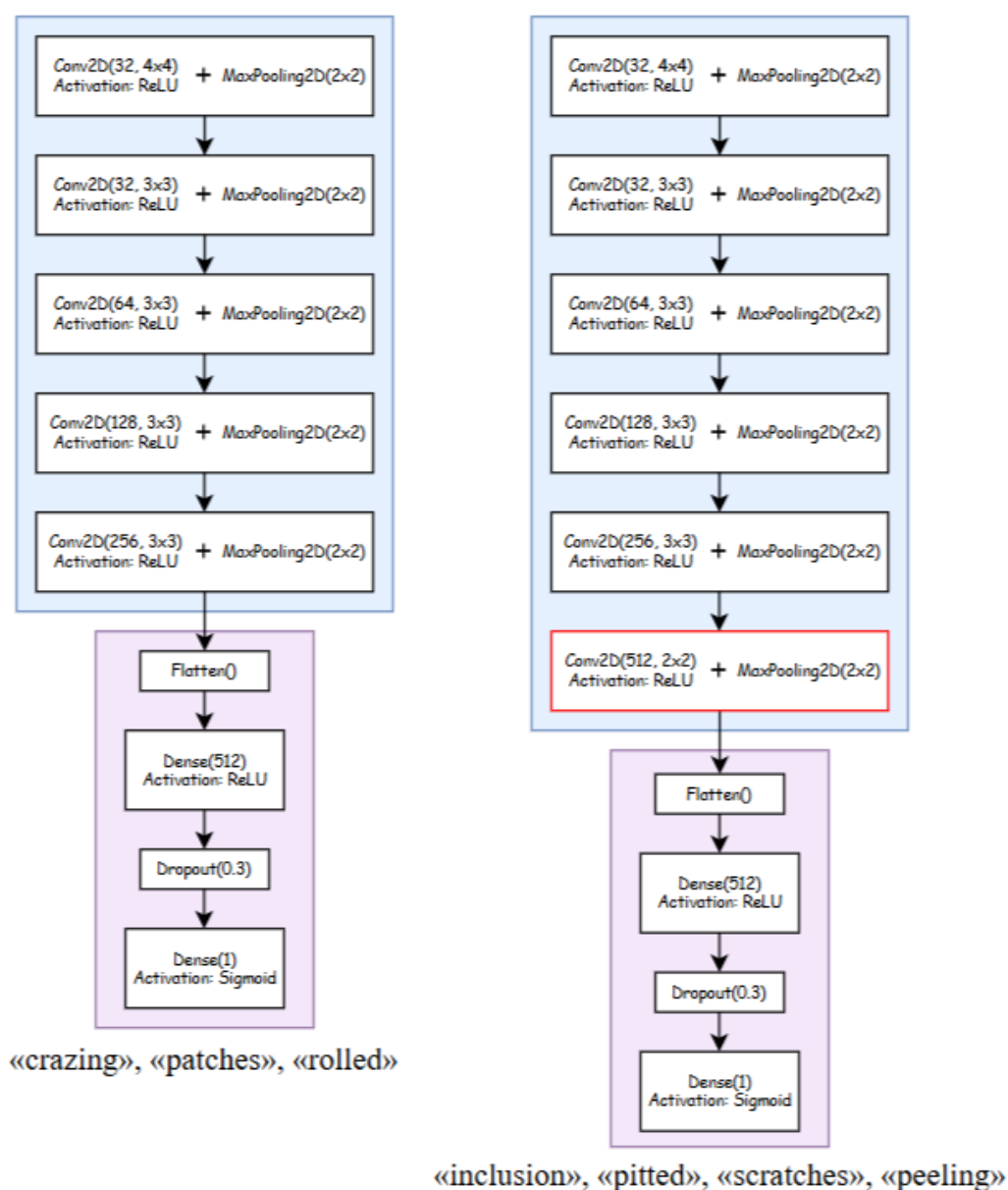


Рисунок 1 – Архитектура предложенных моделей
Figure 1 – Architecture of the proposed models

Данные классов «crazing», «patches» и «rolled» содержали достаточное количество качественных изображений с высокой вариативностью и визуальной четкостью. Модели для них включали в себя:

- пять сверточных слоев (Conv2D) с увеличением числа фильтров ($32 \rightarrow 32 \rightarrow 64 \rightarrow 128 \rightarrow 256$).
- пять слоев максимизирующей подвыборки (MaxPooling2D), которые уменьшают размерность карт признаков в 2 раза.
- два полносвязных слоя в конце, включая Dropout для регуляризации.

Так как при классификации проверочных данных этих классов была получена точность, равная 1,0 для изображений, этого количества слоев было достаточно.

Данные классов «inclusion», «pitted», «scratches» и «peeling» характеризовались недостаточным объемом и низким качеством образцов даже после применения методов аугментации. Так, например, при описанном выше количестве слоев классификация дефекта «inclusion» показывала точность 0,72. В связи с этим были добавлены дополнительный сверточный слой (с 512 фильтрами) и слой максимизирующей подвыборки (MaxPooling2D). Таким образом, классы «inclusion», «pitted», «scratches» и «peeling» включали в себя на два слоя больше.

В ходе обучения моделей использовался оптимизатор Adam (Adaptive Moment Estimation) – метод стохастического градиентного спуска, основанный на адаптивной оценке моментов первого и второго порядка. Для оценки ошибки бинарной классификации использовалась логарифмическая функция потерь (1), где N – количество наблюдений, y_i – фактическая метка класса, p_i – предсказанная вероятность. На основании истинных значений и предсказанных моделью вероятностей вычисляется среднее значение логарифмических потерь для всех примеров. Таким образом, она штрафует модель за отклонения предсказанных вероятностей от истинных меток, достигая минимума при их полном совпадении. Градиенты штрафа указывают, как корректировать параметры модели, чтобы минимизировать ошибку.

$$\text{Log Loss} = \frac{1}{N} \sum_{i=1}^N [y_i \cdot \log(p_i) + (1 - y_i) \cdot \log(1 - p_i)]. \quad (1)$$

Обучение моделей проводилось в течение фиксированного количества итераций, ограниченного 20 эпохами (при размере батча 32 и темпе обучения (learning rate) 0,0001). В качестве функции активации использовалась ReLU для всех сверточных слоев и первого полносвязного слоя, а sigmoid – для последнего слоя, так как она преобразует результат в диапазон от 0 до 1. Также применялась регуляризация с помощью слоя Dropout с коэффициентом 0,3. При этом был реализован механизм ранней остановки на основе функции обратного вызова, который останавливал обучение при достижении точности более 99 процентов на валидационных данных.

Используемые метрики качества классификации. После обучения проводилась валидация моделей на проверочных данных. Каждая модель предсказывала, с какой вероятностью изображение относится к соответствующему классу. Результаты сравнивались, а класс с наибольшим значением вероятности считался итоговым предсказанием. Для анализа качества моделей были выбраны следующие метрики: доля правильных ответов (Accuracy), точность (Precision), полнота (Recall) и F1-мера (F1-Score).

Метрика Accuracy показывает долю правильных ответов модели по всем классам. Для вычисления (2) берется количество верно предсказанных положительных (TP) и отрицательных (TN) примеров, а также количество неверно предсказанных положительных (FP) и отрицательных (FN) примеров.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}. \quad (2)$$

Метрика Precision показывает долю верно предсказанных положительных примеров среди всех предсказанных как положительные (3). В отличие от предыдущей

метрики, в ней учитывается только количество правильно классифицированных положительных объектов (TP) и количество неправильно классифицированных положительных объектов (FP).

$$Precision = \frac{TP}{TP+FP}. \quad (3)$$

Метрика Recall – доля корректно предсказанных положительных примеров среди всех реально положительных (4). Она учитывает количество правильно классифицированных положительных объектов (TP) и неправильно классифицированных отрицательных (FN).

$$Recall = \frac{TP}{TP+FN}. \quad (4)$$

Метрика F1-Score – это гармоническое среднее между Precision и полнотой Recall (5). Она дает сбалансированное представление о работе модели, учитывая при этом ложные срабатывания и пропуски. При этом учитывается оба типа ошибок: когда модель ошибочно приписывает объект к классу, и когда модель пропускает объект класса.

$$F1\ score = 2 \cdot \frac{Precision * Recall}{Precision + Recall}. \quad (5)$$

Результаты

Для оценки качества классификации датасета «NEU» использовалось 180 проверочных изображений. Так как каждое изображение должно пройти через каждую модель, было вычислено время классификации всех изображений для каждого класса. Для вычисления времени, требуемого для классификации всех изображений конкретной моделью, было выполнено 20 независимых измерений. После сбора экспериментальных данных было рассчитано среднее значение времени классификации, которое представляет собой точечную оценку математического ожидания. Для оценки разброса данных и точности полученного среднего значения дополнительно вычислена погрешность, отражающая вариативность измерений. Такой подход обеспечивает достоверность результатов и позволяет учесть возможные колебания времени классификации.

Результаты среднего значения времени и погрешностей можно увидеть в Таблице 1. Оценки были получены с использованием центрального процессора Intel(R) Core(TM) i5-12400F – представителя массового сегмента процессоров. Таким образом, в среднем классификация 180 изображений одной моделью занимает около 290 миллисекунд. Среднее время, требуемое для классификации 180 изображений, занимает около 1,75 секунд, а для классификации одного изображения – около 9,63 миллисекунд.

Таблица 1 – Время классификации всех изображений проверочной выборки
Table 1 – Time required to classify all images from the test sample

Класс	Количество тестовых изображений	Количество слоев	Время для классификации 180 изображений, мс
crazing	30	12	288,85 ± 3,44
inclusion	30	14	291,31 ± 1,78
patches	30	12	292,03 ± 3,61
pitted	30	14	295,15 ± 3,47
rolled	30	12	295,45 ± 3,62
scratches	30	14	291,27 ± 1,48

С учетом заданного количества классов и оборудования возможна обработка до 103 кадра в секунду, что является достаточно хорошим результатом для определения дефектов металлических поверхностей в реальном времени [15]. Это позволяет утверждать, что предложенная модель является производительной даже на относительно слабых вычислительных устройствах, что подтверждает ее пригодность для развертывания в ресурсоограниченных промышленных средах.

В Таблице 2 представлены получившиеся значения каждой метрики для проверочной выборки датасета «NEU».

Таблица 2 – Итоговые оценки качества классификации датасета «NEU»

Table 2 – Resulting classification quality scores on the NEU dataset

Метрика	Оценка
Accuracy	98,33
Precision	98,39
Recall	98,33
F1-Score	98,33

Для более детального и наглядного анализа была построена матрица ошибок (Рисунок 2). Видно, что классы «crazing», «patches», «rolled» и «scratches» полностью корректно определяются, а классы «pitted» и «inclusion» имеют 1 и 2 пропущенных значения соответственно.

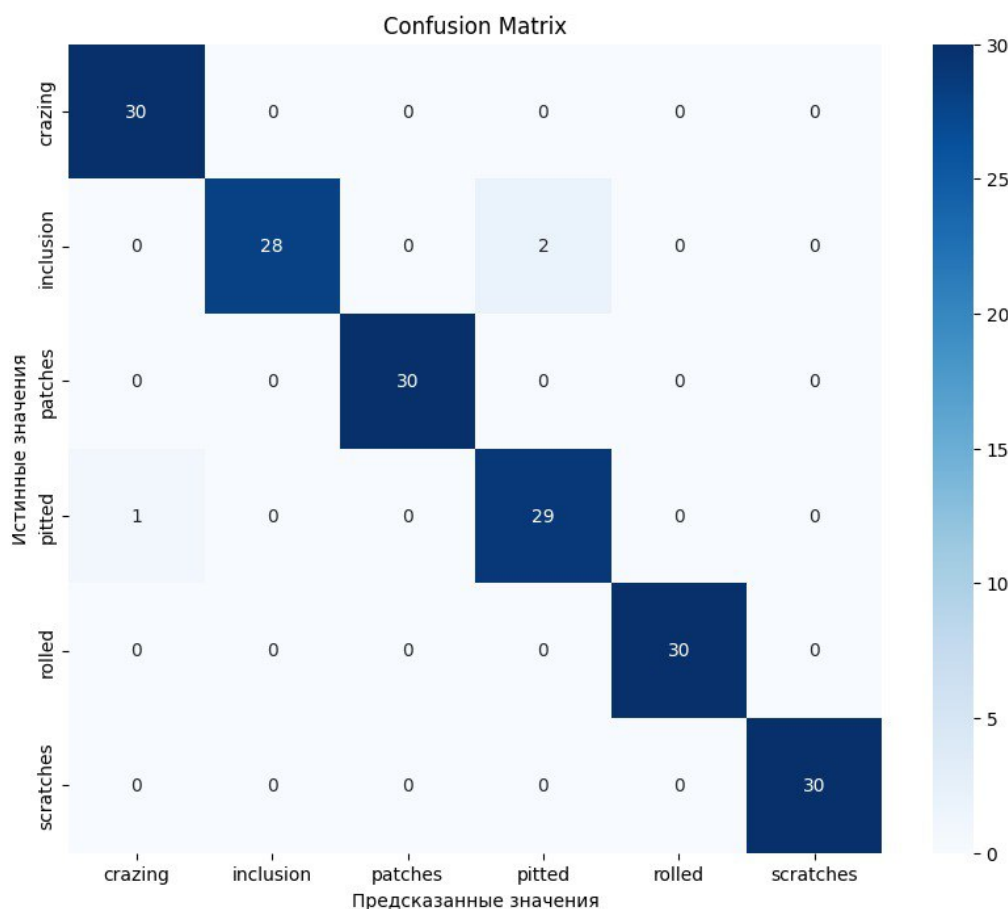


Рисунок 2 – Матрица ошибок по 6 классам

Figure 2 – Confusion matrix for the 6 classes

Для анализа результатов и выявления проблем была проведена визуализация проверочных изображений с выводом вероятности принадлежности к каждому классу, где вероятность больше нуля. Пример для некоторых классов показан на Рисунке 3.

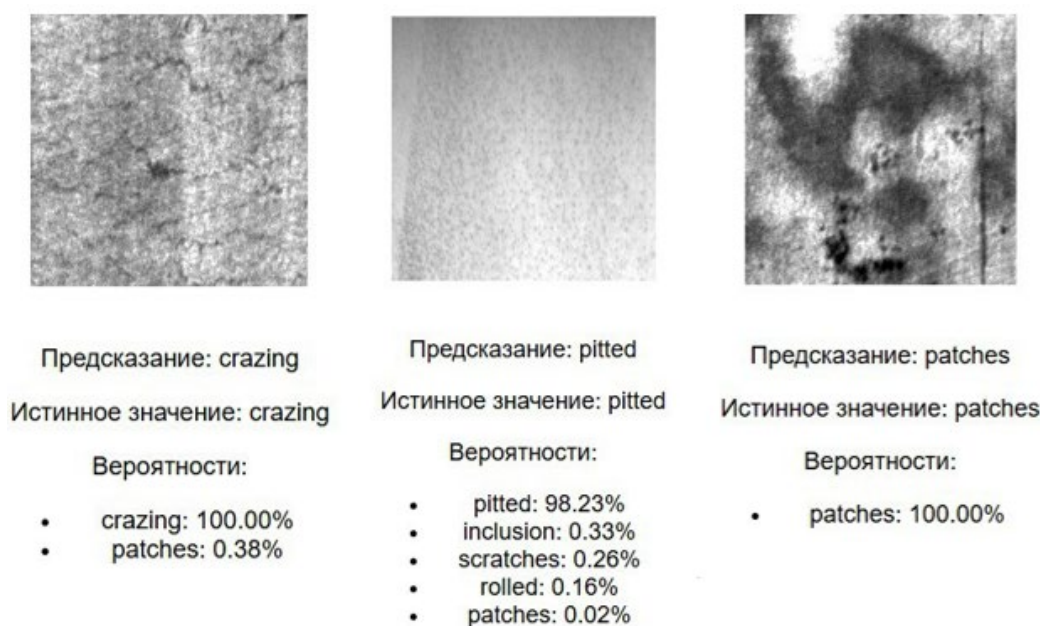


Рисунок 3 – Визуализация предсказаний
Figure 3 – Class prediction visualization

Класс «inclusion» предсказывается с меньшей полнотой, чем остальные, так как модель в двух случаях отнесла изображения этого класса к классу «pitted». Это может быть связано с тем, что фон на изображениях имеет зернистую текстуру, что снижает контраст между дефектом и фоном, и в редких случаях маскирует различия между повреждениями. Ошибочные предсказания показаны на Рисунке 4.

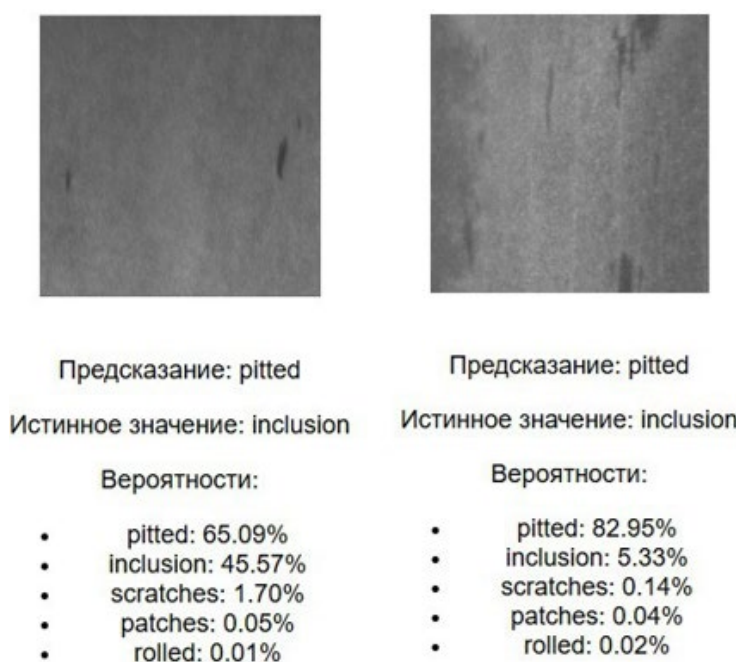
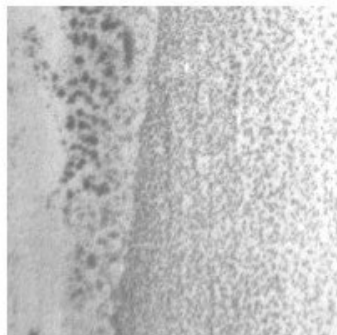


Рисунок 4 – Ошибочные предсказания для класса «inclusion»
Figure 4 – False predictions for "inclusion" class

Модель также отнесла одно изображение класса «pitted» к классу «crazing» (Рисунок 5). Предположительно это связано с тем, что на изображении «pitted» дефект представлен как множество мелких темных точек, расположенных близко друг к другу, что визуально схоже с дефектом «crazing».



Предсказание: crazing

Истинное значение: pitted

Вероятности:

- crazing: 59.25%
- pitted: 0.12%
- patches: 0.05%

Рисунок 5 – Ошибочные предсказания для класса «pitted»

Figure 5 – False predictions for "pitted" class

Далее к текущему датасету был добавлен класс «peeling». Для интеграции нового класса в существующую систему классификации применялся подход «один против всех»: обучалась отдельная модель, специализирующаяся на распознавании дефектов типа «peeling» против всех остальных категорий, тогда как остальные модели не подвергались повторному обучению. После интеграции новой модели общая точность системы снизилась незначительно, что свидетельствует об устойчивости и масштабируемости предложенного решения на основе подхода «one-vs-all». Такое незначительное падение метрики указывает на то, что добавление нового класса не привело к существенной деградации качества распознавания исходных классов. Получившиеся метрики представлены в Таблице 3.

Таблица 3 – Итоговые оценки качества классификации 7 классов

Table 3 – Resulting classification quality scores on the 7-class dataset

Метрика	Оценка
Accuracy	97,14
Precision	97,24
Recall	97,14
F1-Score	97,12

Матрица ошибок для дополненного датасета показана на Рисунке 6.

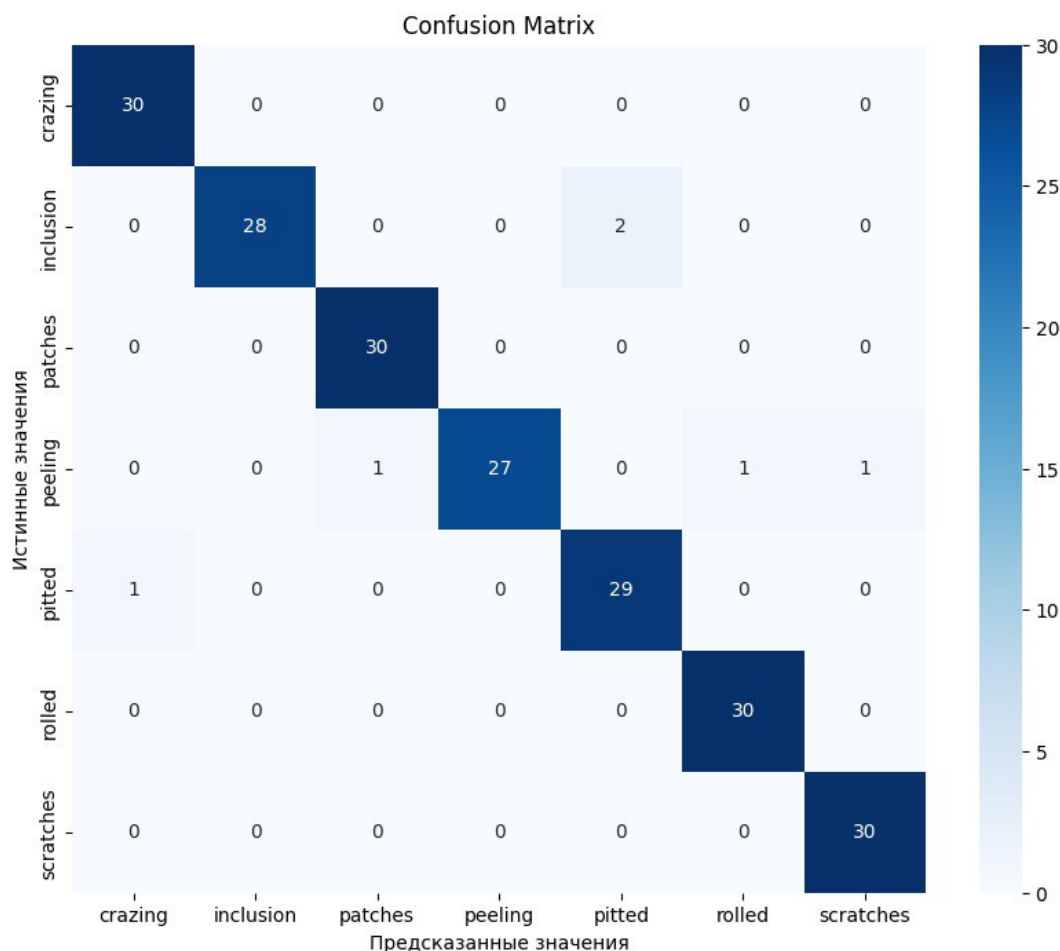


Рисунок 6 – Матрица ошибок по 7 классам
Figure 6 – Confusion matrix for the 7 classes

Анализ матрицы ошибок демонстрирует, что модель сохраняет высокую точность распознавания по всем классам, включая добавленный. Для класса «Peeling» наблюдается высокая полнота, при этом лишь 3 образца были ошибочно отнесены к другим классам. Это свидетельствует о том, что модель успешно выделяет характерные для классов признаки.

3 модели без дополнительных слоев имеют по 3558917 параметров. Объем занимаемой оперативной памяти каждой из них равен около 13,58 Мб. 4 модели с дополнительными слоями имеют по 7489541 параметров, а занимаемый объем оперативной памяти каждой модели составил около 28,57 Мб. Общий объем оперативной памяти, занимаемый всеми моделями, составил 155,02 Мб.

Обсуждение

Для сопоставления результатов рассматривается решение на основе 6 моделей, объединенных с подходом «один против всех», обученных на данных 6 различных типов дефектов из датасета «NEU». Итоговая доля правильных ответов составляет 98,33 %. В Таблице 4 представлено сравнение полученных результатов с другими работами, в которых модели рассматривались в контексте классификации дефектов металлических поверхностей. Для сравнения используются метрики качества классификации, а именно

Accuracy, Precision, Recall и F1-Score. Дополнительно в таблице указано количество классов и используемые датасеты, что позволяет провести сравнительный анализ между архитектурами и подходами.

Таблица 4 – Сравнение результатов
Table 4 – Result comparison

Ссылка на статью	Архитектура	Число классов	Датасет	Accuracy, %	Precision, %	Recall, %	F1-Score, %
Текущая работа	CNN	6	NEU	98,33	98,39	98,33	98,33
[1]	CNN+VGG	6	NEU	99,44	–	–	99
[2]	Xception Inception ResNetV2 InceptionV3	6	NEU	92,22 97,56 97,78	–	–	89,65 97,40 97,76
[3]	SqueezeNet+MSP	6	NEU	100	–	–	–
[4]	MobileNetV2	6	NEU	87	88	87	87
[5]	LMViT	6	NEU-DET	97,22	–	–	–
[13]	InceptionResNetV2 MobileNetV2 DenseNet201	6	NEU	97,33 97,55 98,37	–	–	–
[14]	LCNN	6	NEU	91,67	–	–	–

Оценки доли правильных ответов (Accuracy), точности (Precision) и полноты (Recall) модели, разработанной в рамках текущей работы, достигают 98,33 %, 98,39 % и 98,33 % соответственно. F1-Score составляет 98,33 %, указывая на сбалансированность модели. Полученные результаты значительно выше, чем в работах [4, 14], лучше, чем в работах [2, 5] и лишь незначительно уступают наилучшему результату, представленному в работе [13]. В работах [1, 3] доля правильных ответов выше.

Производительность модели в работе [3] отмечена на уровне 130 кадров в секунду на графическом ускорителе GPU NVIDIA 1080Ti, в то время как в текущей работе получена близкая производительность (103 кадра в секунду) на центральном процессоре.

Разработанная модель обеспечивает достаточно высокие значения по всем ключевым метрикам, что подтверждает ее пригодность для задач классификации дефектов. При этом в данной работе была показана возможность масштабирования модели для включения дополнительных классов, что говорит о практической ценности данного подхода для реальных производственных сценариев, где количество возможных дефектов может быть значительным.

Для минимизации проблемы ошибочной классификации между отдельными классами дефектов в будущем возможно применение ряда усовершенствованных подходов. Одним из перспективных направлений является интеграция механизмов внимания, таких как CBAM (Convolutional Block Attention Module) или SE-блоки (Squeeze-and-Excitation), которые позволяют модели фокусироваться на наиболее информативных участках изображения и тем самым повысят точность модели. Другим направлением может быть использование датасетов большего объема, в том числе включающих больше чем 6 классов дефектов металлических поверхностей.

Заключение

В работе было рассмотрено применение моделей глубокого обучения с подходом «один против всех» для задачи многоклассовой классификации дефектов металлических поверхностей. Использовались данные 6 классов из датасета «NEU Surface Defect Database» и данные 1 класса из «Severstal: Steel Defect Detection». Были предложены две архитектуры сверточных нейронных сетей, используемые для конкретных классов дефектов.

В результате экспериментальной оценки были получены значения доли правильных ответов – 98,33 %, а также точности – 98,93 %, полноты – 98,33 % и F1-меры – 98,33 %. Проведенный анализ показал, что предложенный подход и архитектура нейронных сетей обеспечивают высокую точность, сопоставимую с результатами, получаемыми с помощью иных современных архитектур, таких как CNN+VGG, InceptionResNetV2, MobileNetV2, DenseNet201 и других. Время, затрачиваемое на обработку одного изображения, позволяет использовать полученные модели для классификации с частотой около 103 изображений в секунду. После расширения перечня классов за счет класса из датасета «Severstal: Steel Defect Detection» были получены следующие оценки: доля правильных ответов – 97,14 %, точность – 97,24 %, полнота – 97,14 % и F1-мера – 97,12 %. Эти оценки показали устойчивость и адаптивность предложенного подхода, позволяя эффективно интегрировать новые классы дефектов, в том числе без необходимости полного переобучения уже существующих компонентов модели. Это подтверждает применимость стратегии «один против всех» в условиях динамически расширяемого набора классов.

Результаты исследования могут быть применены в задачах визуального распознавания дефектов металлических поверхностей, включая отрасли, связанные с производством металлопродукции и обработкой металла. Дальнейшие направления исследований могут быть направлены на расширение количества определяемых классов повреждений, улучшение качества классификации в контексте визуально схожих повреждений и оценку ресурсной эффективности систем, в которые может быть интегрирована представленная модель.

СПИСОК ИСТОЧНИКОВ / REFERENCES

1. Lee S.Yo., Tama B.A., Moon S.Ju., Lee S. Steel Surface Defect Diagnostics Using Deep Convolutional Neural Network and Class Activation Map. *Applied Sciences*. 2019;9(24). <https://doi.org/10.3390/app9245449>
2. Юнани Д.Л., Медведев М.В., Кирпичников А.П. Классификация дефектов стали при помощи метода переноса обучения. *Вестник технологического университета*. 2019;22(7):174–178.
Yunani J.L., Medvedev M.V., Kirpichnikov A.P. Classifying Steel Defects Using Transfer Learning. *Herald of Technological University*. 2019;22(7):174–178. (In Russ.).
3. Fu G., Zhang Z., Le W., et al. A Multi-Scale Pooling Convolutional Neural Network for Accurate Steel Surface Defects Classification. *Frontiers in Neurorobotics*. 2023;17. <https://doi.org/10.3389/fnbot.2023.1096083>
4. Han O.C., Kutbay U. Detection of Defects on Metal Surfaces Based on Deep Learning. *Applied Sciences*. 2025;15(3). <https://doi.org/10.3390/app15031406>
5. Ayon S.T.K., Siraj F.M., Uddin J. Steel Surface Defect Detection Using Learnable Memory Vision Transformer. *Computers, Materials & Continua*. 2025;82(1):499–520. <https://doi.org/10.32604/cmc.2025.058361>

6. Wang Sh., Xia X., Ye L., Yang B. Automatic Detection and Classification of Steel Surface Defect Using Deep Convolutional Neural Networks. *Metals*. 2021;11(3). <https://doi.org/10.3390/met11030388>
7. Luo Q., Fang X., Su J., et al. Automated Visual Defect Classification for Flat Steel Surface: A Survey. *IEEE Transactions on Instrumentation and Measurement*. 2020;69(12):9329–9349. <https://doi.org/10.1109/TIM.2020.3030167>
8. Li Q., Luo Zh., Chen H., Li Ch. An Overview of Deeply Optimized Convolutional Neural Networks and Research in Surface Defect Classification of Workpieces. *IEEE Access*. 2022;10:26443–26462. <https://doi.org/10.1109/ACCESS.2022.3157293>
9. Prihatno A.T., Utama I.B.K.Yo., Kim Ju.Yo., Jang Y.M. Metal Defect Classification Using Deep Learning. In: *Proceedings of the 2021 Twelfth International Conference on Ubiquitous and Future Networks (ICUFN), 17–20 August 2021, Jeju Island, Korea*. IEEE; 2021. P. 389–393. <https://doi.org/10.1109/ICUFN49451.2021.9528702>
10. Konovalenko I., Maruschak P., Brezinová J., Viňáš J., Brezina J. Steel Surface Defect Classification Using Deep Residual Neural Network. *Metals*. 2020;10(6). <https://doi.org/10.3390/met10060846>
11. Konovalenko I.V., Maruschak P.O. Classification of the Surface Technological Defects in Rolled Metal Products with the Help of a Deep Neural Network. *Materials Science*. 2021;56(6):779–788. <https://doi.org/10.1007/s11003-021-00495-5>
12. Zhu P., Dai Ju., Chang H., Xu Ya., Zhang Zh. DenseNet Network-Based Surface Defect Detection Algorithm for Strip Steel. *International Journal of Frontiers in Engineering Technology*. 2022;4(10):34–40. <https://doi.org/10.25236/IJFET.2022.041006>
13. Chaudhry N.Ja., Khan M.B., Iqbal M.Ja., Yasir S.M. Modeling & Evaluating the Performance of Convolutional Neural Networks for Classifying Steel Surface Defects. *Journal of Artificial Intelligence*. 2022;4(4):245–259. <https://doi.org/10.32604/jai.2022.038875>
14. Mamun A.M.A., Hossain R., Sharmin M.M. Detection and Classification of Metal Surface Defects Using Lite Convolutional Neural Network (LCNN). *Material Science & Engineering International Journal*. 2024;8(3):72–75.
15. Dabkowski P., Gal Ya. Real Time Image Saliency for Black Box Classifiers. In: *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems, 04–09 December 2017, Long Beach, CA, USA*. New York: Curran Associates, Inc.; 2017. P. 6967–6976.

ИНФОРМАЦИЯ ОБ АВТОРАХ / INFORMATION ABOUT THE AUTHORS

Сосновская Владислава Евгеньевна, студентка, **Vladislava E. Sosnovskaya**, Student, MIREA – МИРЭА – Российский технологический университет, Москва, Российская Федерация. Russian Technological University, Moscow, the Russian Federation.
e-mail: vlada.sosnovskaya@gmail.com
ORCID: [0009-0000-7953-6577](https://orcid.org/0009-0000-7953-6577)

Гаврилова Алла Дмитриевна, студентка, **Alla D. Gavrilova**, Student, MIREA – Russian МИРЭА – Российский технологический университет, Москва, Российская Федерация. Technological University, Moscow, the Russian Federation.
e-mail: gavrilowaa2004@gmail.com
ORCID: [0009-0004-6855-8067](https://orcid.org/0009-0004-6855-8067)

Волкова Елизавета Алексеевна, студентка, **Elizaveta A. Volkova**, Student, MIREA – МИРЭА – Российский технологический университет, Москва, Российская Федерация. Russian Technological University, Moscow, the Russian Federation.
e-mail: lizabat15@gmail.com
ORCID: [0009-0009-4936-8037](https://orcid.org/0009-0009-4936-8037)

Котилевец Игорь Денисович, старший преподаватель кафедры КБ-14 «Цифровые технологии обработки данных», МИРЭА – Russian Technological University, Moscow, the Russian Federation. Department KB-14 "Digital Data Processing Technologies, MIREA – Russian Technological University, Moscow, the Russian Federation.
e-mail: ikotilevets@gmail.com
ORCID: [0000-0003-0433-3999](https://orcid.org/0000-0003-0433-3999)

Ильин Дмитрий Юрьевич, кандидат технических наук, доцент кафедры КБ-14 «Цифровые технологии обработки данных», МИРЭА – Russian Technological University, Moscow, the Russian Federation. Dmitry Y. Ilin, Candidate of Engineering Sciences, Associate Professor at the Department KB-14 "Digital Data Processing Technologies", MIREA – Russian Technological University, Moscow, the Russian Federation.
e-mail: i@dmityilin.com
ORCID: [0000-0002-0241-2733](https://orcid.org/0000-0002-0241-2733)

Статья поступила в редакцию 18.07.2025; одобрена после рецензирования 12.10.2025; принята к публикации 22.10.2025.

The article was submitted 18.07.2025; approved after reviewing 12.10.2025; accepted for publication 22.10.2025.