

УДК: 697.3:004.85
DOI: 10.48612/NewsKSUAE/76.7
EDN: FUGVEE



Аппроксимация результатов численного моделирования узла «тройник-колено» с помощью сверточной нейронной сети

А.М. Зиганшин¹, В. И. Шакирзянов²

¹Казанский государственный архитектурно-строительный университет,

²ООО «Испытательный центр «Стройэксперт»», г. Казань, Российская федерация

Аннотация: Численное моделирование методами вычислительной гидродинамики элементов систем вентиляции хотя и считается сейчас основным инструментом исследования и проектирования, имеет существенный недостаток - затрата большого количества вычислительных ресурсов. Это замедляет и исследование, и внедрение результатов в практику проектирования. Целью исследования является применение сверточной нейронной сети для аппроксимации полей скорости и давления для ускорения расчётов отрывных течений и аэродинамического сопротивления вентиляционного узла типа «тройник-колено». Для этого разработан метод преобразования данных из Ansys Fluent в формат пригодный для обучения нейронной сети; созданы данные с информацией о граничных условиях геометрии и параметрах течения воздуха; реализована архитектура сверточной нейронной сети; обучена модель на имеющихся данных и проведена оценка точности аппроксимации на контрольной выборке конфигураций узла.

Результаты. Разработан подход, при котором на первом этапе результаты численного моделирования преобразуются в матричный формат и используются для глубокого обучения. При этом входные данные включают: бинарные маски, поля знаковых расстояний и граничные условия. В качестве архитектуры была выбрана сеть с остаточными и пропускными связями. Далее обучение проводилось на расширенном наборе данных с использованием нормализации и комбинированной функции потерь, учитывающей градиенты полей. После было проведено исследование, и настройка гиперпараметров и в итоге получена модель нейросети, которая обеспечивает достаточно высокую точность прогнозирования компонент скорости: коэффициент детерминации превышает 0,98, а средняя ошибка составляет 8–12 %. Точность аппроксимации поля давления несколько ниже и составила около 27,5 %.

Выводы. Разработанный метод позволяет быстрее и с достаточной степенью достоверности находить распределение давления и компонент скорости в узлах вида «тройник-колено». Полученные результаты могут быть использованы при расчете и проектировании энергоэффективных элементов систем вентиляции.

Ключевые слова: сверточная нейронная сеть; аппроксимация полей течения; численное моделирование; вычислительная гидродинамика; вентиляционные узлы; проектирование инженерных систем

Для цитирования: Зиганшин А.М., Шакирзянов В.И. Аппроксимация результатов численного моделирования узла «тройник-колено» с помощью сверточной нейронной сети // Известия КГАСУ, 2026, № 2 (76), с. 100-121, DOI: 10.48612/NewsKSUAE/76.7, EDN: FUGVEE

Approximation of numerical simulation results for a “tee-sharp elbow” unit using a convolutional neural network

A.M. Ziganshin, V.I. Shakirzyanov

¹ Kazan State University of Architecture and Engineering,
Kazan, Russian Federation

² LLC “Stroyekspert” Testing Center, Kazan, Russian Federation

Abstract. Although numerical simulation of ventilation system components using computational fluid dynamics (CFD) methods currently considered as a primary research and design tool, it has a significant drawback: it consumes significant computational resources. This slows both research and the implementation of results in design practice. The objective of this study is to apply a convolutional neural network to approximate velocity and pressure fields to accelerate the calculation of separated flows and aerodynamic drag in a tee-elbow ventilation unit. To achieve this goal, a method was developed for converting data from Ansys Fluent into a format suitable for neural network training; data was generated with information on the boundary conditions of the geometry and airflow parameters; a convolutional neural network architecture was implemented; a model was trained using the available data; and the approximation accuracy was assessed using a test sample of unit configurations.

Results. An approach has been developed in which, in the first stage, numerical simulation results are converted into a matrix format and used for deep learning. The input data includes binary masks, signed distance fields, and boundary conditions. A network with residual and throughput connections was chosen as the architecture. Training was then performed on an expanded dataset using normalization and a combined loss function that accounts for field gradients. Subsequently, a study was conducted and hyperparameters were tuned, resulting in a neural network model that provides a fairly high accuracy in predicting velocity components: the determination coefficient exceeds 0.98, and the average error is 8–12%. The accuracy of the pressure field approximation is somewhat lower, at approximately 27.5%.

Conclusions. The developed method allows for faster and more reliable determination of the pressure distribution and velocity components in tee-elbow units. The results obtained can be used in the calculation and design of energy-efficient ventilation system components.

Keywords: convolutional neural network; flow field approximation; numerical modeling; computational fluid dynamics; ventilation units; engineering systems design.

For citation: Ziganshin A.M., Shakirzyanov V.I. Approximation of numerical simulation results for a “tee-sharp elbow” unit using a convolutional neural network // News of KSUAE, 2026, № 2 (76), p. 100-121, DOI: 10.48612/NewsKSUAE/76.7, EDN: FUGVEE

1. Введение

Моделирование процессов течения воздуха или смеси газов в различных элементах систем вентиляции имеет большое значение для сферы проектирования инженерных систем. Вклад потерь на местных сопротивлениях в общие потери давления может превышать 85% [1]. Помимо этого, отдельные элементы системы могут взаимно влиять друг на друга, увеличивая или уменьшая общие потери давления [2]. На сегодняшний день проводится множество исследований на тему взаимного влияния местных сопротивлений в системах вентиляции [3–5], однако, из-за большого множества различных конфигураций узлов на это тратится большое количество времени и вычислительных ресурсов.

Авторы статьи [6] утверждают, что одного роста вычислительной мощности компьютеров недостаточно, чтобы сделать моделирование численными методами быстрым и общедоступным инструментом для решения задач отопления, вентиляции и кондиционирования воздуха. Эту же проблему подтверждают авторы работ [7,8] говоря о том, что применение методов вычислительной гидродинамики затруднено из-за

ограниченности пропускной способности памяти и хаотичного доступа к ней для пересчета соседних ячеек сетки геометрии. Для решения данной проблемы в работе рассматривается комбинированный подход: аппроксимация готовых результатов численного моделирования конфигураций узла с помощью сверточной нейронной сети.

Аналогичный гибридный подход используется в [9] для создания модели на базе состязательного автоэнкодера (Adversarial Autoencoder, AAE), который обучается на результатах CFD моделирования методом крупных вихрей (Large Eddy Simulation, LES) и позволяет эффективно предсказывать потоки воздуха, концентрации CO_2 и распределение температуры в помещении с людьми. Далее нейронная сеть, дающая увеличение скорости предсказания полей на порядок по сравнению с CFD и используется для управления инженерными системами в реальном времени.

Попытка обобщить результаты эволюции использования машинного обучения (Machine Learning, ML) для моделирования течений, на момент выхода статьи (июнь 2025 г.), проведена в [10]. Авторы обзора, опираясь на проанализированные работы предлагают такое хронологическое деление:

- Поколение 1: классические суррогаты – Kriging [11], полиномиальная регрессия и метод поверхности отклика (Response Surface Modeling) [12]. Такие модели строят математическую аппроксимацию (например, с помощью полиномов или пространственной интерполяции) между входными параметрами задачи (например, геометрией помещения) и выходными данными (например, температурой или скоростью в точке). Работают хорошо только для задач с малым числом параметров (оптимизация профиля крыла [13] или предсказание поля скорости и турбулентности при обтекании нескольких зданий, простой формы [14]). Для 3D-гидродинамики и сложных задач чаще всего непригодны из-за "проклятия размерности" (Curse of Dimensionality) – экспоненциальном росте количества обрабатываемых данных, при увеличении количества исследуемых параметров.
- Поколение 2: «Чистый Data-Driven ML» – сверточные нейронные сети (Convolutional Neural Network, CNN), сети с долгой краткосрочной памятью (Long Short-Term Memory, LSTM), метод «Случайный лес» (Random Forest). Используя эти алгоритмы сеть учится отображать "вход -> выход" напрямую из CFD-данных [15,16]. Проблема такого подхода в том, что это по сути "черный ящик". Если подать на вход геометрию, которой не было в обучающей выборке, сеть выдаст нефизичную картину, например, при задании условий воздухораспределения при перемешивающей вентиляции для модели обученной на вытесняющей [17].
- Поколение 3: Модели пониженного порядка (Reduced Order Models, ROM) - класс математических подходов, предназначенных для существенного упрощения сложных вычислительных систем. Например, решение о течении воздуха в помещении или в канале полученное методом вычислительной гидродинамики (CFD) описывается десятками параметров, которые связаны физическими уравнениями для каждой из сотен тысяч или миллионов расчетных ячеек. Это сильно замедляет решение такой системы уравнений. Использование ROM позволяет заменить (сжать) эту систему уравнений, с использованием алгоритма «Собственная ортогональная декомпозиция» (Proper Orthogonal Decomposition, POD)[18]. Например в работе [9] вместо того, чтобы обучать нейросеть предсказывать значения скорости, температуры и CO_2 для всех 54 363 узлов трехмерной сетки, авторы сначала применили метод POD. В результате исходное большое пространство данных было сжато до ~120 главных коэффициентов (координат в низкоразмерном пространстве). Нейросеть обучалась предсказывать эволюцию во времени именно этих 120 чисел, а не миллионов.
- Поколение 4: физически-информированные нейросети (Physics-Informed, PINN) – встраивают фундаментальные законы физики (например, уравнения Навье-Стокса) непосредственно в функцию потерь нейросети [19,20]. Это радикально решает проблему обобщаемости (Generalization) [21]: в отличие от чисто статистических (data-driven) моделей, PINN опирается и ограничена физикой процесса, поэтому способна корректно экстраполировать данные на незнакомые геометрии или

граничные условия, не выдавая физически абсурдных результатов. Однако этот подход порождает новые вычислительные трудности. Функция потерь PINN содержит несколько конкурирующих слагаемых (ошибка по реальным данным и невязка физических уравнений), веса которых крайне сложно сбалансировать. Это делает процесс оптимизации нестабильным, часто приводя к застреванию алгоритма в локальных минимумах или полному отсутствию сходимости [20].

Кроме того, авторы обзора выделили три уже сейчас используемых направления использования машинного обучения (Machine Learning, ML) при CFD моделировании. Создание расчетных сеток, которое сейчас, при помощи классических инструментов, занимает до 80% времени работы инженера, особенно для сложных случаев расчетной геометрии. Это связано и с необходимостью подбора областей сгущения сетки, что нейросети (например, Implicit Geometry Neural Networks) помогают выполнять быстрее, сокращая до 85% времени препроцессинга [10,22,23]. Нейросети могут встраиваться и в этап самого численного решения, например решение уравнения Пуассона для давления, которое занимает основную долю времени в несжимаемых средах [24] или расчет тензоров рейнольдсовых напряжений в RANS-моделях (Reynolds-Averaged Navier–Stokes) турбулентности [25,26]. Также существенное ускорение достигается при анализе полученного решения на этапе постпроцессинга. Например Вместо ручного поиска зон застоя воздуха или вихрей, сверточные сети (CNN) делают это автоматически [27,28], а также занимаются супер-разрешением (восстановлением детального 3D-поля по грубой сетке) [29,30].

Ранее авторами [31] было проведено решение тестовой задачи течения в канале с препятствием при помощи сверточной нейронной сети на базе архитектуры U-Net. В результате были найдены такие параметры модели, при которых прогноз полей скорости и давления давал приемлемую точность при небольшой выборке данных: коэффициент детерминации (R^2) модели составил 0,926, средневзвешенная ошибка в процентах (WAPE – Weighted Average Percentage Error) по трем полям равна 23,94%. Исследование показало, что свёрточные нейронные сети могут выявить закономерности в физике течения среды (отрыв потока и изменение скорости вдоль стенок), приблизившись к точности решения численных методов, решающие трудоемкие уравнения Навье–Стокса.

В статье [32] авторы сравнили работу сверточных нейронных сетей U-Net и ResUNet (Residual U-Net) с их доработанными версиями FSC-UNet (FSC – Fourier Skip Connection) и FSC-ResUNet с помощью внедрения нейронного оператора Фурье (FNO – Fourier Neural Operator) в связь между энкодером и декодером. Прогноз полей осуществлялся для обтекания высотных зданий при различных направлениях ветра. Результаты показали, что предсказания FSC-ResUNet хорошо согласуются с решениями численных методов, R^2 при этом приблизился к 0,999, а WAPE в среднем 3,51%. Сверточная сеть без оператора Фурье ResUNet показала следующие результаты: R^2 приблизился к 0,99, WAPE в среднем 18,32%. Авторы отметили, что добавление остаточных связей к U-Net (то есть ResUNet) значительно улучшает ее возможности в прогнозе, позволяя модели лучше улавливать нелинейные изменения в полях течения.

Несмотря на выдающиеся результаты FSC-ResUNet, модель тратит больше времени на обучение при больших разрешениях из-за сложности алгоритма преобразования Фурье. В работе [33] авторы заявляют, что ResUNet обучается примерно в 1,5–7,5 раз быстрее чем с использованием нейронных операторов Фурье (Fourier Neural Operator, FNO), а использование гибридных архитектур с нейронным оператором Фурье, расположенном в каждом блоке модели (например UFNO - U-Net Enhanced Fourier Neural Operator) приводит к увеличению времени обучения в 2,5–5,0 раз по сравнению с FNO.

Развитие методов машинного обучения для гидродинамики не ограничивается только свёрточными архитектурами. Например, в исследовании [34] проведен анализ возможностей и ограничений физически-информированных нейронных сетей (PINN – Physics-Informed Neural Network) для моделирования: течения Куэтта и Пуазейля, вязкоупругих сред (модель Максвелла) и др. Авторы заявляют, что PINN могут дополнять и расширять традиционные методы вычислительной гидродинамики, но на их взгляд все еще сохраняются проблемы со стабильностью обучения таких моделей и обработкой граничных условий.

В свою очередь, авторы статьи [35] провели сравнение моделей-аппроксиматоров на основе сверточных нейронных сетей (CNN – Convolutional Neural Network) и графовых нейронных сетей (GNN – Graph Neural Network) на наборе данных аэродинамики автомобиля. Метод, основанный на CNN, при меньших затратах на обучение достиг точности прогнозирования в 78% со средней абсолютной ошибкой в 2,3. Метод, основанный на GNN, имел более высокую ошибку в 3,8 и точность прогнозирования в 77%. Результаты показали, что оба метода могут помочь в нахождении оптимальных форм геометрии, прежде чем приступить к трудоемкому моделированию.

В работе [36] предлагается два различных способа представления геометрии неправильной формы и погруженных тел для прогнозирования полей скорости. Оба метода были применены в данной работе. Простой метод заключается в преобразовании геометрии в бинарное изображение, где 1 – область течения, 0 – отсутствие решения. Более сложный метод заключается в вычислении поля знаковых расстояний (SDF – Signed Distance Field) в области течения, см. например [31].

Для обучения любой нейронной сети необходимо составить функцию потерь. Функция потерь – это формула, оценивающая насколько хорошо или плохо модель предсказывает данные. Распространенными нормами потерь являются среднеквадратичная ошибка (MSE – Mean Squared Error) и средняя абсолютная ошибка (MAE – Mean Absolute Error). В прошлом исследовании [31] MSE показала себя достаточно хорошо, так как штрафует нейронную сеть за малейшие ошибки в предсказании, однако поля на выходе получались не гладкими из-за особенностей сверточной обработки. В работе [37] авторами предлагается функция потерь с градиентом полей для повышения точности и эффективности обучения физически-информированных нейронных сетей (PINN – Physics-Informed Neural Network). Функция использует информацию о градиенте величин (скорости и давления), что повышает точность и эффективность обучения.

Из проведенного обзора следует, что применение нейронных сетей в связке с методами вычислительной гидродинамики является хорошей альтернативой численному моделированию множества конфигураций геометрий и граничных условий. Наиболее сбалансированным выбором архитектуры является ResUNet с физически-информированной функцией потерь, обеспечивающая гладкость полей.

Целью данного исследования является решение задачи аппроксимации полей компонент скорости и давления узла «тройник-колено» на основе ResUNet.

Для достижения цели решаются следующие задачи:

- разработать метод преобразования данных из Ansys Fluent в формат пригодный для обучения нейронной сети;
- создать данные с информацией о граничных условиях геометрии и параметрах течения воздуха;
- реализовать выбранную архитектуру сверточной нейронной сети;
- обучить модель на имеющихся данных и оценить точность аппроксимации на контрольной выборке конфигураций узла.

2. Материалы и методы

Исходные данные. Исследование проводится для численно решенных комбинаций геометрических параметров узла и граничных условий на ответвлениях в двух направлениях течения – приточный и вытяжной тройник. Изменяемыми параметрами являются расстояние между тройником и коленом, и отношение расходов воздуха в ответвлениях. Геометрия расчетной области приведена на рисунке 1.

Колено расположено на сборном участке тройника. При численном решении длины каналов l_a и l_c приняты исходя из обеспечения выравнивания профиля скорости и равны $40b$ и $20b$ соответственно, где $b = 0,1$ м – ширина канала. Исследование проводится при изменении расстояния l_b между коленом и тройником в следующем диапазоне: $1b$; $3b$; $5b$; $10b$. Отношения расходов воздуха r на ветвях тройника: $0,9/0,1$; $0,7/0,3$; $0,3/0,7$; $0,1/0,9$ и смоделированы при помощи граничного условия «pressure outlet» на границах AB/II с установкой разных значений избыточного давления, EF – приточная или вытяжная граница – граничное условие «velocity inlet» со скоростью равной 40 м/с для случая

приточного узла или -40 м/с – для вытяжного. На остальных границах задана непроницаемая стенка «wall».

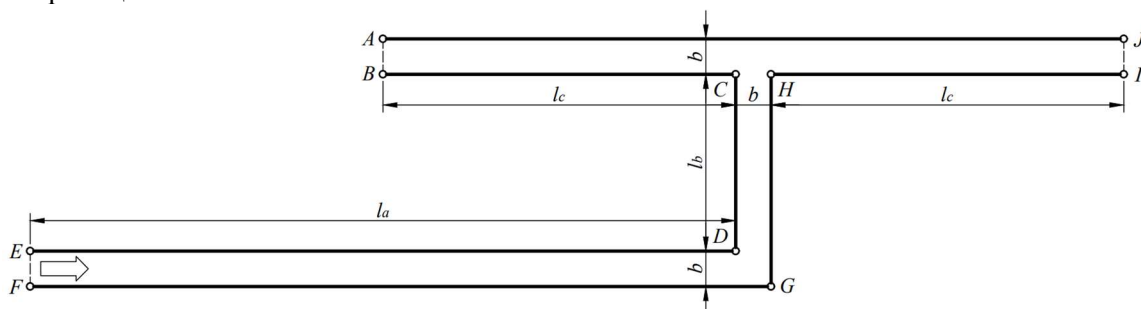


Рис. 1. Геометрия узла «колено-тройник» для приточного варианта (иллюстрация авторов)

Fig. 1. Geometry of the unit «elbow-tee» supply case (illustration by the authors)

Итоговое число расчетных случаев составляет 32 примера и определяется комбинацией четырех вариантов расстояния l_b , четырех отношений расходов на ответвлениях r и двумя направлениями течения воздуха в узле. Максимальный размер поля составил 1,2х6,1 м.

Преобразование данных. Решенные поля выгружались из Ansys Fluent в виде ASCII файла, где к каждой координате ячейки относилось одно значение компонент скорости и давления. Экспорт данных осуществлялся с применением официальной библиотеки PyFluent от Ansys Fluent для Python в цикле прохода каждого расчетного случая. Далее данные со значениями преобразовывались в класс DataFrame библиотеки pandas, округлялись до 6 знака после запятой и сортировались по координатам x , y . Вывод такой таблицы в терминал показан на рисунке 2.

При помощи метода `pivot_table` библиотеки pandas производилась переконфигурация данных из табличного вида в матричный, где строки координат выполняли функцию индексации элемента создаваемой матрицы, а значением элемента являлась величина скорости или давления в данной точке.

	x-coordinate	y-coordinate	x-velocity	y-velocity	pressure
0	0.001	0.001	-39.216297	-2.983222	-2211.804688
1	0.001	0.003	-39.879482	-4.380151	-2118.319092
2	0.001	0.005	-40.019806	-4.738197	-2057.729980
3	0.001	0.007	-40.103413	-4.856638	-1990.147949
4	0.001	0.009	-40.131176	-4.776580	-1953.563721
...	...	...	...	...	...
1829995	6.099	1.191	-3.665529	-0.076539	-8.164707
1829996	6.099	1.193	-3.665529	-0.076539	-8.164707
1829997	6.099	1.195	-3.665529	-0.076539	-8.164707
1829998	6.099	1.197	-3.665529	-0.076539	-8.164707
1829999	6.099	1.199	-3.665529	-0.076539	-8.164707

[1830000 rows x 5 columns]

Рис. 2. Вывод программы после обработки данных (иллюстрация авторов)

Fig. 2. Program output after data processing (illustration by the authors)

Таким образом формировались три матрицы (1):

$$\Phi = \begin{bmatrix} \varphi_{11} & \dots & \varphi_{1j} & \dots & \varphi_{1n} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \varphi_{i1} & \dots & \varphi_{ij} & \dots & \varphi_{in} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \varphi_{m1} & \dots & \varphi_{mj} & \dots & \varphi_{mn} \end{bmatrix} \quad (1)$$

где Φ – поле компонент скорости или давления;

φ – значение параметра в конкретной координате;

m, n – количество элементов по оси x, y соответственно (2, 3).

$$m = \frac{y_{max}}{\Delta y} = \frac{1,2}{0,002} = 600 \quad (2)$$

$$n = \frac{x_{max}}{\Delta x} = \frac{6,1}{0,002} = 3050 \quad (3)$$

где $\Delta x = \Delta y$ – шаг сетки геометрии, равный 0,002 м.

Далее было произведено кадрирование матриц к размеру 1050x1050, исключая зоны выравнивания потока. Центрирование проводилось по четырем ячейкам в середине участка между тройником и коленом (рис. 3).

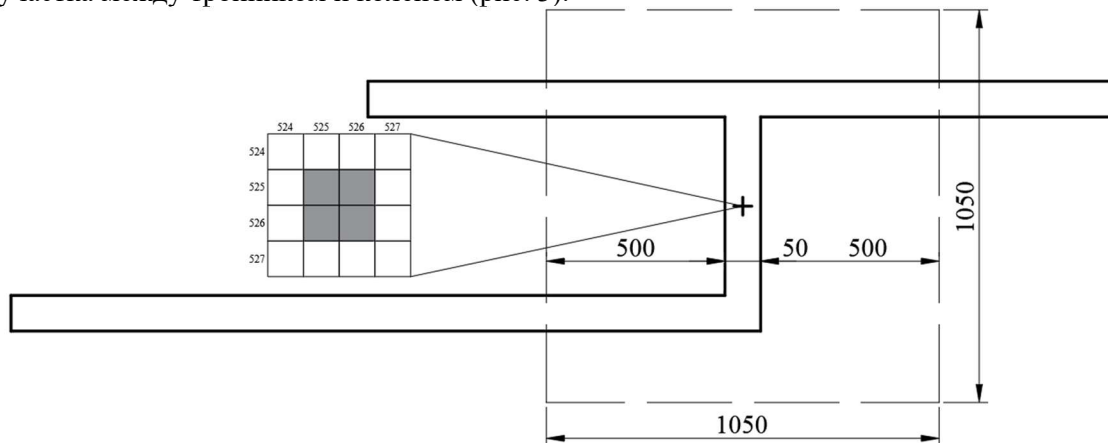


Рис. 3. Изменение размера поля (иллюстрация авторов)
Fig. 3. Changing the size of field (illustration by the authors)

Результат кадрирования размера матриц приведен на рисунке 4.

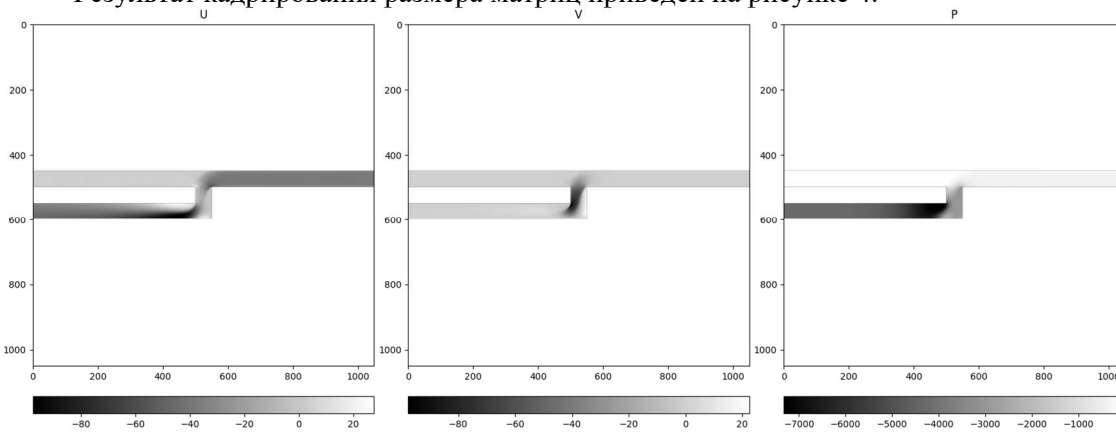


Рис. 4. Результат кадрирования полей U (слева), V (по центру) и P (справа) (иллюстрация авторов)
Fig. 4. Result of cropping the fields U (left), V (mid) and P (right) (illustration by the authors)

Следующим этапом было создание признаков карт с информацией о граничных условиях и геометрии узлов для эффективной работы нейронной сети. Например, из-за неправильной формы узла, которая вписана в прямоугольную матрицу, возникают пустые области не имеющие решения. Для того, чтобы модель могла различить отсутствие решения в ячейке от реального значения скорости или давления была создана бинарная маска, хранящая информацию о расчетной области течения Ω (4).

$$M = \begin{cases} 1, & m_{ij} \in \Omega \\ 0, & m_{ij} \notin \Omega \end{cases} \quad (4)$$

где m_{ij} – элемент матрицы маски.

Далее кодируется информация о граничных условиях. Вводятся два канала: SDF1 и SDF2 (SDF – Signed Distance Field). Алгоритм создания SDF1 описан в работе [9]. Он является мерой расстояния от стенок канала до оси течения. Поле SDF2 хранит информацию о том, как далеко расположены выходы от входа в узел. Оно создано с помощью алгоритма обхода поля (BFS – Breadth-First Search). Итоговые матрицы приведены на рисунке 5.

Последним каналом является поле со значениями скорости по модулю на трех границах узла. Данная матрица будет содержать информацию о начальной скорости на входе в узел и отношении скоростей на его выходах. Значения заполнены шириной в 2 элемента на границах, вся остальная область течения равна единице (рис. 6).

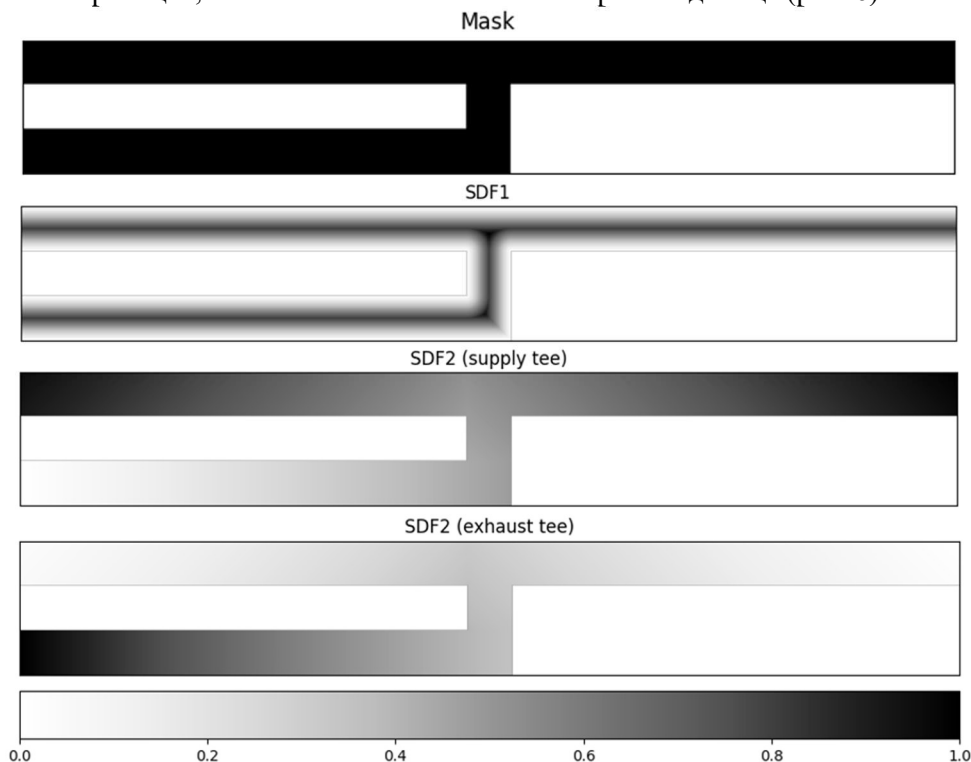


Рис. 5. Бинарная маска (первое поле), расстояние от стенок канала до оси течения (второе), расстояние от входа в узел до его выходов для приточного тройника (третье) и для вытяжного (четвертое) (иллюстрация авторов)

Fig. 5. Binary mask (first field), distance from the walls to the flow axis (second), distance from unit inlet to outlet for supply tee (third) and exhaust tee (fourth) (illustration by the authors)

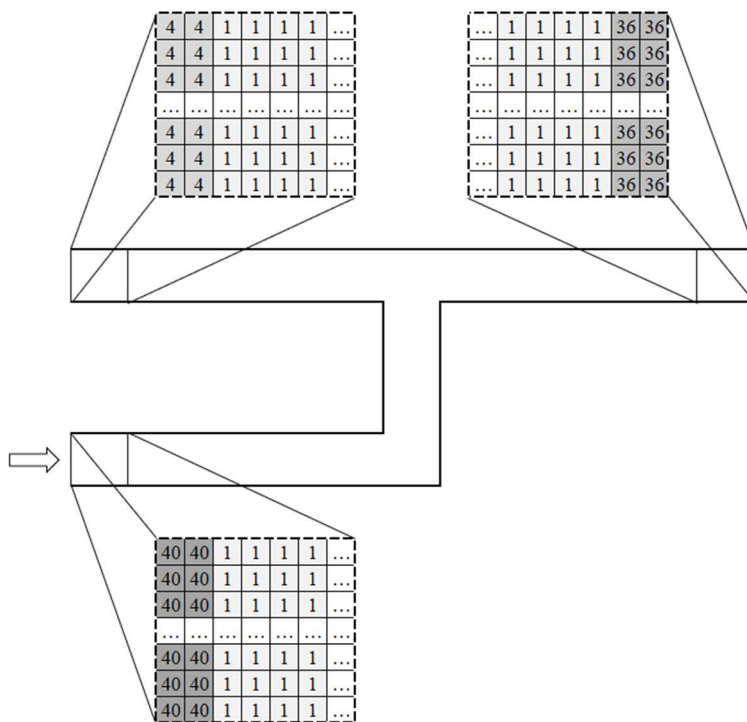


Рис. 6. Поле с информацией о величине скорости на границах узла (иллюстрация авторов)

Fig. 6. Velocity magnitude at unit borders (illustration by the authors)

Итоговая размерность данных о геометрии и граничных условиях составила (4, 1050, 1050), а размерность решенных полей из Ansys Fluent равна (3, 1050, 1050) для одной конфигурации узла. Первый пакет является входными данными для обучения, а второй это «ответы» для нейронной сети, с которыми и будет проводиться вычисление ошибки предсказания.

Аугментация. При малом числе данных (32 примера) и большой размерности матриц (1050, 1050) нейронная сеть не может в полной мере запомнить детальные структуры течения. Для этого применяют аугментацию данных – искусственное увеличение набора примеров путем отражений и поворотов матриц. Выполнено отражение полей по оси x , y с изменением знака компонент скоростей в зависимости от направления. Поле давления является скаляром, поэтому при отражениях знак остается неизменным. Результаты аугментации для U (компоненты скорости по оси x) приведены на рисунке 7, аналогичные результаты получаются и для остальных полей. Общее количество конфигураций после аугментации возросло в 4 раза и составило 128 примеров.

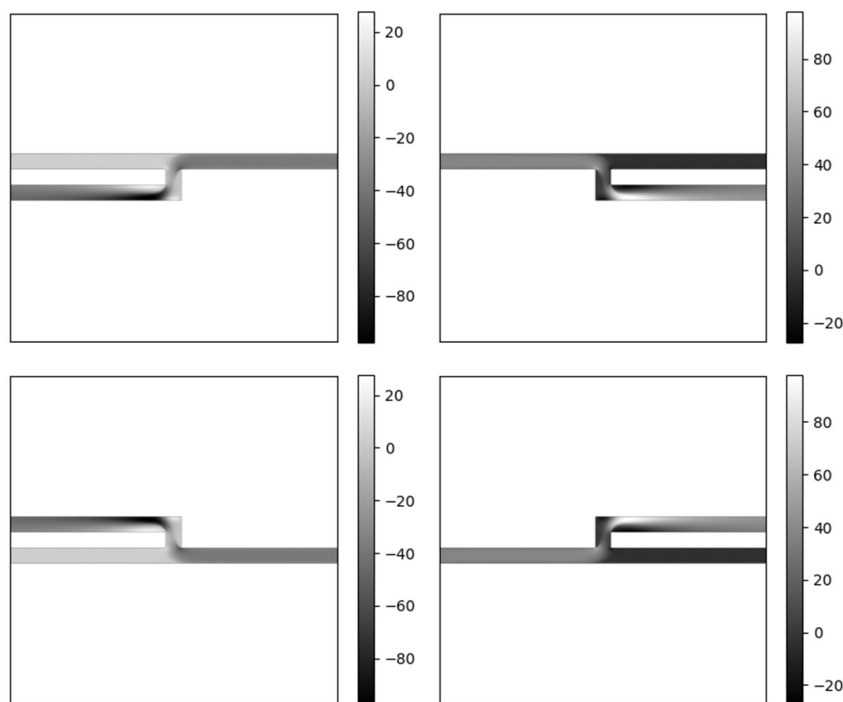


Рис. 7. Результат аугментации поля скорости U (иллюстрация авторов)
Fig. 7. Result of the augmentation of the velocity field U (illustration by the authors)

Архитектура модели. В основе работы лежит глубокая сверточная нейронная сеть, построенная по принципу Encoder-Decoder с пропуск-связями (skip connections) и с остаточными связями внутри блока (residual connections) – ResUNet.

Создается сверточный слой, состоящий из операций: Conv2d, WeightNorm, BatchNorm2d и Activation. Из сверточных слоев создается остаточный блок (residual block) с дополнительной функцией активации на выходе и связью между входом и выходом. Входной тензор перед тем, как пройти через первый слой, копируется и отправляется в конец блока (остаток). После прохождения оригинальным тензором через слои происходит поэлементное суммирование его с остатком и к результату применяется функция активации.

Из остаточных блоков создается энкодер. Он содержит в себе уровни: набор блоков (определяемый с помощью параметра layers) и операции MaxPool2d. Количество уровней определяется длиной списка фильтров. После прохождения тензором через уровень, количество его каналов изменяется на величину, определяемой той же позицией в списке фильтров, что и номер уровня. Далее операция MaxPool2d понижает размерность матрицы вдвое. На последнем остаточном блоке тензор копируется и передается на соответствующий уровень в декодер через пропуск-связь.

Декодер выполняет функцию восстановления размерности и по структуре он симметричен энкодеру. Отличия состоят в том, что в остаточном блоке внутри сверточного слоя вместо Conv2d используется обратная операция ConvTranspose2d, а вместо MaxPool2d применяется MaxUnPool2d. Также перед подачей тензора в остаточный блок он конкатенируется с копией тензора, пришедшего из энкодера по пропуск-связи.

Детальная структура ResUNet представлена на рисунке 8. Сплошной линией обозначено основное направление движения данных по сети, пунктирной – остаточные связи внутри блока, а штриховой – пропуск-связи между энкодером и декодером.

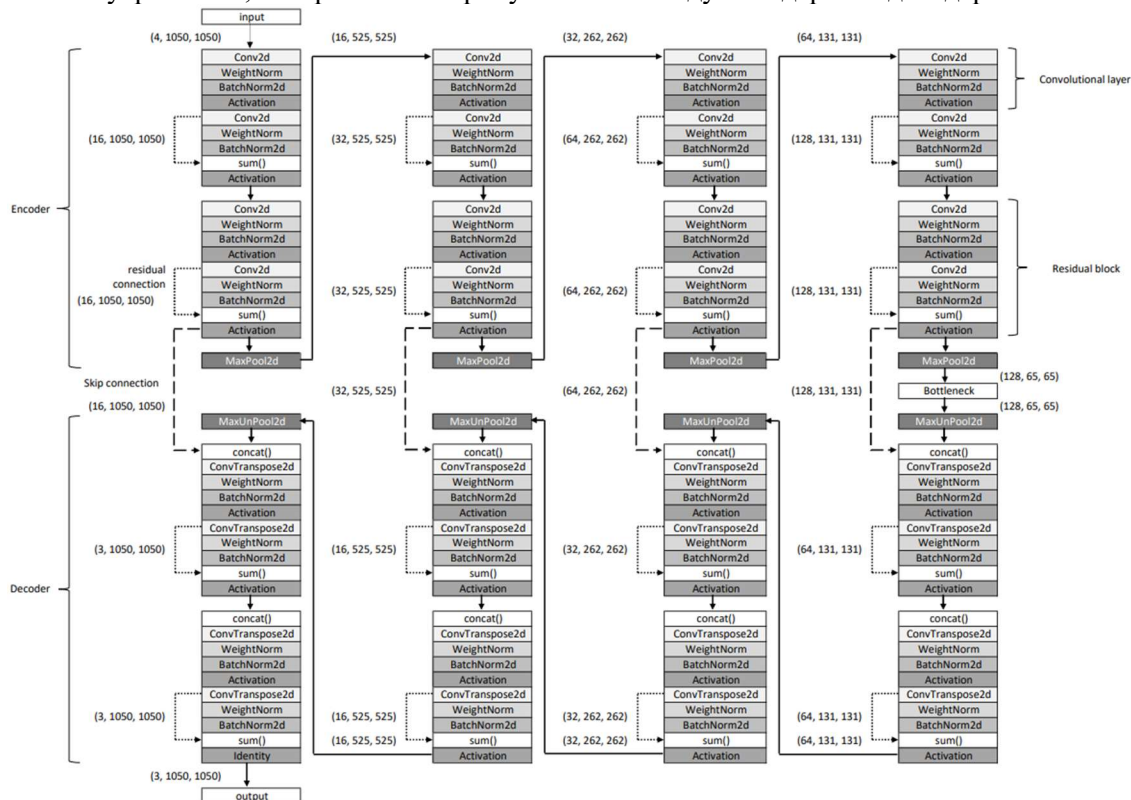


Рис. 8. ResUNet при filters = [16, 32, 64, 128] (иллюстрация авторов)

Fig. 8. ResUNet for filters = (source: 16, 32, 64, 128) (illustration by the authors)

Обучение. Для того, чтобы модель научилась находить закономерности в течении необходимо разделить данные на тренировочную, валидационную и тестовую выборки. Тренировочная выборка (Train dataset) служит для непосредственного обучения и запоминания моделью закономерностей в данных. Нейронная сеть многократно обрабатывает одни и те же примеры уменьшая ошибку предсказания. Из-за явления переобучения модели (заучивания примеров без способности обобщать новые данные) вводится выборка для валидации. Она нужна для контроля обучения и напрямую не влияет на оптимизацию настроек нейронной сети (обновление весов и их градиентов). Сравнение ошибок предсказания на первой и второй выборках позволяет найти сходимости – выход на плато, где ошибка осциллирует в некотором интервале. Тестовая выборка используется однократно после завершения обучения как контрольная проверка модели.

В данной работе примеры были разделены вручную. В тестовую выборку отправились 2 примера, выходящие за пределы обучения и 2 примера, находящиеся в середине всех данных по типу геометрии и граничным условиям. На валидацию и обучение данные разбиты в отношении 30/70. Подробное распределение примеров приведено в таблице 1.

В тестовой выборке примерами для оценки способности модели к экстраполированию являются приточный тройник с расстоянием 10b и отношением

расходов 0,9/0,1, а также вытяжной тройник с расстоянием $1b$ и отношением расходов 0,1/0,9. Два других примера нужны для оценки интерполяции.

Таблица 1

Распределение примеров по выборкам

Тип	Отношения расходов, r	Расстояние между местными сопротивлениями, l_b			
		1b	3b	5b	10b
Приточный	0,1/0,9	<i>Val</i>	Train	Train	Train
	0,3/0,7	Train	<i>Val</i>	Train	Train
	0,7/0,3	Train	Train	Test	<i>Val</i>
	0,9/0,1	Train	Train	<i>Val</i>	Test
Вытяжной	0,1/0,9	Test	<i>Val</i>	Train	Train
	0,3/0,7	<i>Val</i>	Test	Train	Train
	0,7/0,3	Train	Train	<i>Val</i>	Train
	0,9/0,1	Train	Train	Train	<i>Val</i>

В отличие от классического подхода с равномерным распределением всех классов, набор данных, где находится контрольный пример *Test*, имеет всего 3 обучающих примера *Val/Train*. Такое распределение вводится для проверки устойчивости модели к разреженности данных, что не редко бывает в исследованиях, основанных на численном моделировании.

В глубоких нейронных сетях ненормализованные данные негативно влияют на расчет ошибки и ведут к нестабильному обучению. В гидродинамике поля скорости и давления часто имеют разный масштаб, поэтому в построении модели нормализация являлась важным этапом подготовки данных. В работе принята Z-score нормализация (стандартная оценка), которая приводит поля к осредненному диапазону (5):

$$\Phi_{norm} = \frac{\Phi - \bar{\Phi}}{\sigma_{\Phi}} \quad (5)$$

где Φ – поле компонент скорости и давления;

$\bar{\Phi}$ – среднее значение поля в тренировочной выборке данных;

σ_{Φ} – то же, стандартное отклонение.

Расчет среднего значения и стандартного отклонения проводился по тренировочной выборке, во избежание утечки данных при валидации и тестировании модели. Так как нейронная сеть обучалась на нормализованных данных, предсказание на выходе было денормализовано, чтобы вернуть настоящий масштаб полей.

Для функции потерь принята MSE (Mean Squared Error / Среднеквадратичная ошибка) норма (L2) и рассчитывалась следующим образом (6):

$$L = L_{field} + w_{grad} \cdot L_{grad} \quad (6)$$

где L_{field} – общая ошибка предсказания полей (7);

L_{grad} – общая ошибка градиентов между предсказанными и истинными полями (9);

w_{grad} – вклад ошибки градиентов в общую функцию потерь.

Общая ошибка предсказания полей (7):

$$L_{field} = \frac{1}{3}(L_U + L_V + L_P) \quad (7)$$

где U, V, P – индексы ошибки предсказания поля для компонент скорости и давления (8).

Среднеквадратичная ошибка MSE предсказания поля (8):

$$L_y = \frac{1}{n} \sum_M (\hat{\Phi}_{norm} - \Phi_{norm})^2 \quad (8)$$

где $\hat{\Phi}_{norm}$ – значение в точке предсказанного поля в нормализованном пространстве;

Φ_{norm} – то же, в точке истинного поля;

n – количество ячеек, вошедших в область течения (сумма значений бинарной маски M);

M – область течения воздуха (матрица бинарной маски).

Общая ошибка градиентов полей (9):

$$L_{grad} = \frac{1}{3}(L_{grad U} + L_{grad V} + L_{grad P}) \quad (9)$$

Взвешенная ошибка градиентов по полю (10):

$$L_{grad\phi} = \frac{1}{2} \left(\frac{L_{grad_x\phi}}{\sigma^2_{grad_x\phi}} + \frac{L_{grad_y\phi}}{\sigma^2_{grad_y\phi}} \right) \quad (10)$$

где $\sigma^2_{grad_{x,y}\phi}$ – дисперсия матрицы градиентов истинных полей Φ .

Для корректного расчета матриц градиентов использовались ненормализованные данные (в реальном масштабе). Так как ошибка по полю (7) считалась в нормализованном пространстве, то было необходимо уравнивать вклад ошибки двух компонент функции потерь. Нормой потерь является среднеквадратичная ошибка, следовательно в формуле 10 происходило деление на дисперсию значений истинных полей градиентов, а не на стандартное отклонение.

Среднеквадратичная ошибка градиентов по осям x, y (11):

$$L_{grad_{x,y}\phi} = \frac{1}{n} \sum_M (grad_{x,y}\hat{\phi} - grad_{x,y}\phi)^2 \quad (11)$$

где $grad_{x,y}\hat{\phi}$ – градиент в точке предсказанного поля по осям x, y ;
 $grad_{x,y}\phi$ – то же, для истинного поля.

Расчет матриц градиентов для предсказанных и истинных данных осуществлялся с помощью метода центральных разностей (12, 13):

$$grad_x\phi \approx \frac{\phi_{i,j+1} - \phi_{i,j-1}}{2h}, \quad (12)$$

$$grad_y\phi \approx \frac{\phi_{i+1,j} - \phi_{i-1,j}}{2h} \quad (13)$$

где $h = 1$ – расстояние между соседними ячейками в матрице.

Поиск гиперпараметров. Был проведен поиск оптимальных параметров для ResUNet таких как: скорость обучения (learning rate), размер ядра фильтра (kernel size), количество фильтров (filters), количество слоев в остаточном блоке (layers), нормализация весов (weight norm), нормализация примеров в одной эпохе (batch norm), количество примеров в одной эпохе (batch size) и вес вклада ошибки градиентов полей в общую функцию потерь w_{grad} . Все проведенные опыты достигли сходимости.

Для воспроизводимости исследований было зафиксировано начальное значение для генератора случайных чисел (seed). Генератор используется для стартовой инициализации весов и алгоритмов работы операций внутри сверточной нейронной сети. Загрузка данных, расчет ошибок и построение сверточной нейронной сети было проведено с использованием библиотеки PyTorch.

Обучение проводилось на видеокарте RTX 5080 с применением формата чисел с плавающей запятой bfloat16. На заключительных опытах расчет одной эпохи в среднем занимал 8,4 с или 3,5 часа на полное обучение модели. После расчета каждой эпохи сравнивалась ошибка на валидации: если она была меньше, чем на предыдущей эпохе, то настройки модели сохранялись в отдельный файл.

Для каждого прогноза по тестовой выборке вычислялся коэффициент детерминации и средневзвешенная абсолютная ошибка в процентах по формулам 14, 15.

$$R^2 = 1 - \frac{\sum_M (\phi - \hat{\phi})^2}{\sum_M (\phi - \bar{\phi})^2} \quad (14)$$

$$WAPE = 100\% \cdot \frac{\sum_M |\phi - \hat{\phi}|}{\sum_M |\phi|} \quad (15)$$

где φ – значение в точке истинного поля U, V, P ;

$\hat{\varphi}$ – то же, предсказанного поля;

$\bar{\varphi}$ – среднее значение истинного поля U, V, P ;

M – область течения воздуха.

Коэффициент детерминации показывает общую дисперсию предсказанных значений от истинных и является вспомогательным параметром при оценке точности модели. Основной метрикой является средняя ошибка WAPE, показывающая относительную ошибку, нормированную на сумму фактических значений.

3. Результаты и обсуждение

Для выявления нелинейных закономерностей течения нейронная сеть должна обладать достаточной глубиной (большое общее количество скрытых слоев), однако неконтролируемое увеличение числа фильтров ведет к утрате информации на последнем слое энкодера, так как на каждом уровне происходит уменьшение размерности матриц вдвое и соответственно потеря части информации. При проведении опытов лучше всего показала следующая комбинация параметров, которая обеспечила хороший баланс между точностью и стабильностью обучения. Архитектура модели построена на основе ядра размером 7×7 (`kernel_size=7`), что позволяет захватывать более крупные пространственные паттерны уже на начальных слоях, не теряя при этом в вычислительной эффективности. Сеть состоит из двух основных блоков (`layers=2`), в которых количество фильтров последовательно удваивается — от 8 до 128 (`filters=[8, 16, 32, 64, 128]`). Такая прогрессия позволяет модели сначала выделять простые локальные признаки, а затем комбинировать их в более сложные и абстрактные представления. В качестве функции активации использована классическая ReLU (Rectified Linear Unit) — быстрая, устойчивая к проблеме затухающих градиентов и хорошо совместимая с пакетной нормализацией.

Процесс обучения настроен на длительную и точную подстройку весов: модель обучается в течение 1500 эпох со скоростью обучения 0,001 — значением, которое обеспечивает надёжную сходимость без риска «проскакивания» оптимума. Размер мини-пакета (`batch_size=4`) намеренно сделан очень маленьким: это добавляет шум в оценку градиента, что работает как неявная регуляризация и помогает модели лучше обобщать, избегая переобучения на шумных данных. Для стабилизации распределения активаций и ускорения сходимости включена пакетная нормализация BatchNorm2d, тогда как WeightNorm оставлен отключённым, чтобы не дублировать функции нормализации и сохранить архитектуру проще. В совокупности эти настройки формируют компактную, но выразительную модель, способную эффективно обучаться даже на ограниченных или зашумлённых данных при условии тщательного контроля за валидационной метрикой в ходе длительного тренировочного цикла.

После подбора основных параметров модели проверялась величина вклада ошибки градиентов полей в общую функцию потерь в диапазоне от 0 до 1 с шагом 0,2 (табл. 2).

Таблица 2

Сравнение R^2 и WAPE для 6 значений w_{grad}

w_{grad}	R^2			WAPE		
	U	V	P	U	V	P
0,0	0,952	0,943	0,859	13,75%	19,52%	27,46%
0,2	0,917	0,981	0,868	13,89%	13,40%	27,35%
0,4	0,981	0,982	0,879	8,78%	11,50%	27,54%
0,6	0,980	0,975	0,878	8,98%	12,00%	26,03%
0,8	0,937	0,968	0,856	18,08%	16,03%	29,38%
1,0	0,970	0,977	0,792	10,49%	15,60%	34,61%

Как видно по таблице 2 наилучшие результаты по всем трем полям получены при $w_{grad} = 0,4$. При этом R^2 для компонент скоростей находится на высоком уровне ($> 0,98$), а ошибка не более 12%. Поле давления P имеет ошибку свыше 26%, а R^2 менее 0,88. При значении $w_{grad} = 0,6$ точность модели слегка падает, но остается на приемлемом уровне. Дальнейшее изменение w_{grad} вне интервала $[0,4; 0,6]$, приводит к снижению обобщающей

способности модели. Также демонстрируется высокая чувствительность поля давления к изменению w_{grad} : после 0,8 его ошибка растет быстрее чем у компонент скорости.

Незначительное изменение метрик при $w_{grad} = 0,4$ и $w_{grad} = 0,6$ может говорить о существовании локального минимума или максимума для WAPE и R^2 . Оптимальное значение w_{grad} вероятно лежит в этом интервале и требует уточнения. Для этого проведем 4 дополнительных опыта с разбиением масштаба на 0,4; 0,44; 0,48; 0,52; 0,56; 0,6 (табл. 3, рис. 9).

Таблица 3

Сравнение R^2 и WAPE в интервале $w_{grad} \in [0,4; 0,6]$

w_{grad}	R^2			WAPE		
	U	V	P	U	V	P
0,4	0,981	0,982	0,879	8,78%	11,50%	27,54%
0,44	0,980	0,979	0,877	9,57%	14,65%	28,23%
0,48	0,939	0,951	0,847	16,25%	21,23%	32,16%
0,52	0,981	0,968	0,858	8,72%	14,15%	26,04%
0,56	0,943	0,952	0,802	14,45%	19,10%	29,62%
0,6	0,980	0,975	0,878	8,98%	12,00%	<u>26,03%</u>

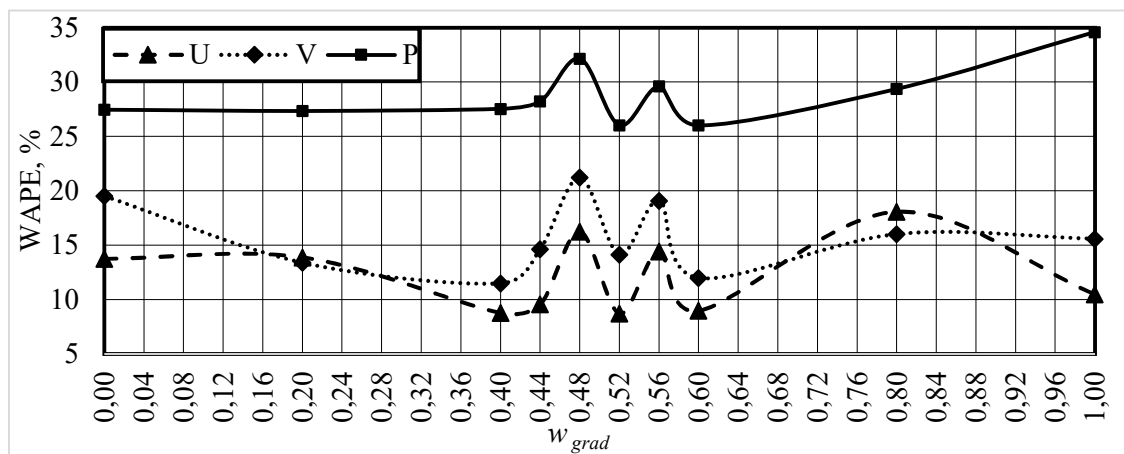


Рис. 9. Зависимость WAPE от веса w_{grad} для полей скорости U, V и давления P
(иллюстрация авторов)

Fig. 9. Weighted Average Percentage Error of velocity U, V and pressure P fields as a function of weight w_{grad} (illustration by the authors)

По таблице 3 и графику 9 видно, что интервал $[0,4; 0,6]$ содержит 3 локальных минимума (в точках 0,4; 0,52; 0,6) и 2 локальных максимума (в точках 0,48 и 0,56) для функций WAPE и R^2 . Дополнительные опыты показали, что оптимальными значениями для w_{grad} остаются 0,4 и 0,6, так как при их проведении упала точность модели, по сравнению с предыдущими результатами.

Предпочтительным выбором является $w_{grad} = 0,4$, значение которого обеспечивает наиболее высокую обобщающую способность модели для всех полей: $R^2_U = 0,981$; $R^2_V = 0,982$; $R^2_P = 0,879$; $WAPE_U = 8,78\%$; $WAPE_V = 11,50\%$; $WAPE_P = 27,54\%$.

Таблица 4

Результаты аппроксимации

Пример	R^2			WAPE		
	U	V	P	U	V	P
В-1b-0,1/0,9	0,911	0,896	0,697	15,41%	30,68%	40,14%
В-3b-0,3/0,7	0,990	0,988	0,971	5,56%	10,91%	11,98%
П-5b-0,7/0,3	0,991	0,996	0,723	7,60%	7,38%	39,50%
П-10b-0,9/0,1	0,995	0,989	0,889	6,53%	9,32%	20,01%

Как видно по таблице 4, модель показывает сопоставимые результаты как на интерполяции (1 и 4 примеры), так и на экстраполяции (2 и 3 примеры). При условии разреженности данных (3 обучающих примера на один класс конфигураций) ResUNet

довольно точно предсказывает поля компонент скоростей U , V при средней ошибке в 8–12%, что позволяет ее рекомендовать для расчета функций тока поля и определения зон отрывов потока. Наиболее сложным каналом в обучении является поле давления P , ошибка прогноза которого в среднем равна 26,04%. Это подтверждает и график ошибок на обучении (рис. 10): в отличие от компонент скорости U , V ошибка по давлению P имеет повышенную зашумленность на протяжении всего процесса обучения, а большой разрыв между Train Loss и Val Loss говорит о недообучении – модель достигла предела обобщающей способности для данного канала. Результаты работы сверточной нейронной сети представлены в табл. 4 и на рис. 11–13.

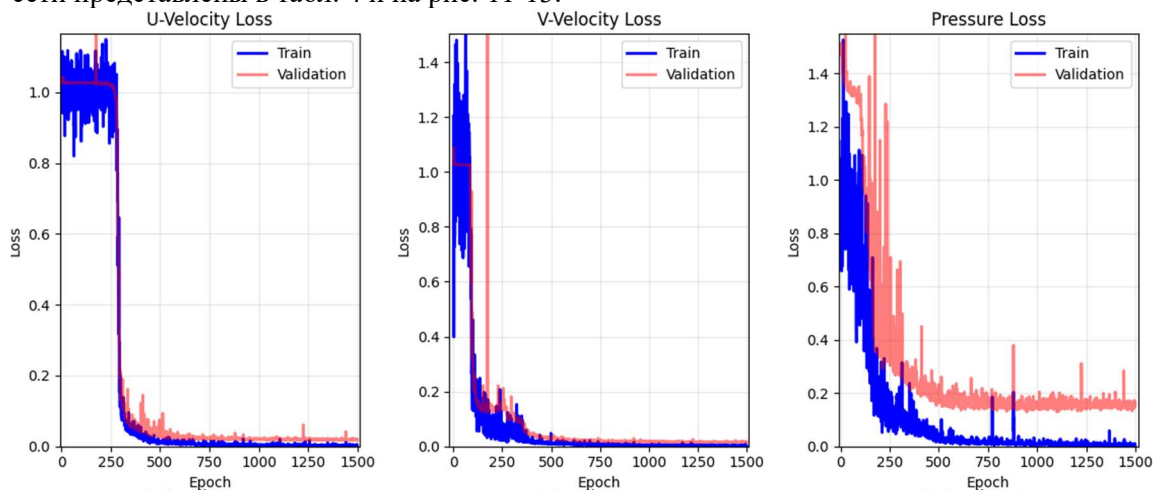


Рис. 10. График ошибок для полей скорости U , V и давления P в процессе обучения нейронной сети (иллюстрация авторов)

Fig. 10. Loss of the velocity U , V and pressure P fields during neural network training (illustration by the authors)

Как видно по рисункам 11–13 модель точно воспроизвела структуру потока. Имеются локальные ошибки в пристеночных зонах и в местах отрыва потока. Пример с $l_b = 1b$ по сравнению с другими геометриями имеет нестандартное распределение скоростей (рис. 11–12), поэтому в данном случае ResUNet справилась хуже: $WAPE_U = 15,41\%$ и $WAPE_V = 30,68\%$. На рисунке 13 видно, что диапазоны значений давления сильно смещены (на 500–1000 Па) и нет резких перепадов, имеющиеся в оригинальном поле – модель сглаживает все неровности.

Для увеличения точности прогноза поля давления P необходимо увеличение набора данных либо тонкая настройка модели: изменение скорости обучения на поздних этапах обучения, подбор функции активации или добавление отдельного декодера для данного канала в архитектуру нейронной сети.

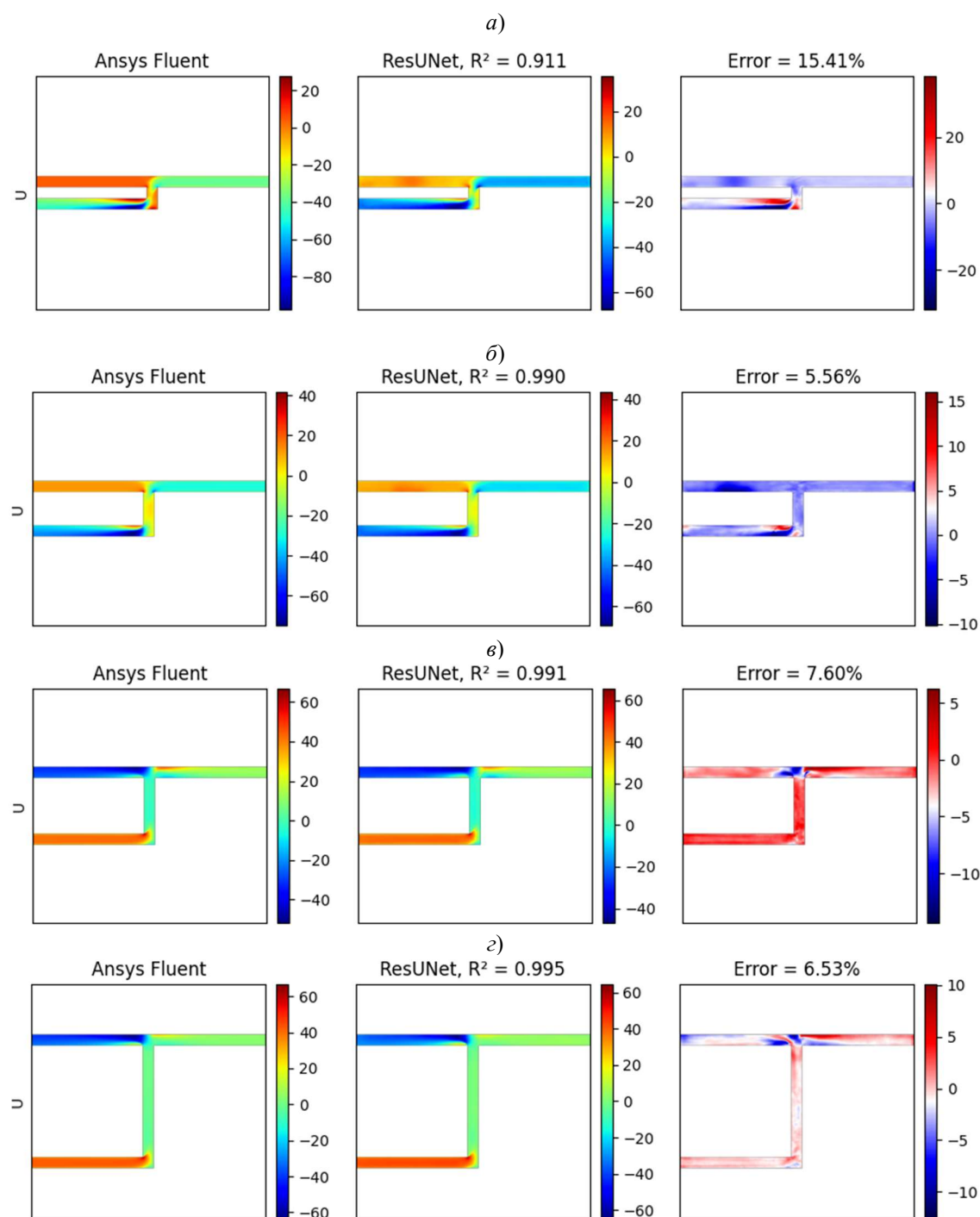


Рис. 11. Результаты аппроксимации поля компоненты скорости U для четырех конфигураций узла с вытяжным тройником: а) $l_b = 1b$ и $r = 0,1/0,9$; б) $l_b = 3b$ и $r = 0,3/0,7$; в) $l_b = 5b$ и $r = 0,7/0,3$; г) $l_b = 10b$ и $r = 0,9/0,1$ (иллюстрация авторов)

Fig. 11. Results of approximation of the field of the velocity component U for four configurations of a node with an exhaust tee: а) $l_b = 1b$ and $r = 0.1/0.9$; б) $l_b = 3b$ and $r = 0.3/0.7$; в) $l_b = 5b$ and $r = 0.7/0.3$; г) $l_b = 10b$ and $r = 0.9/0.1$ (illustration by the authors)

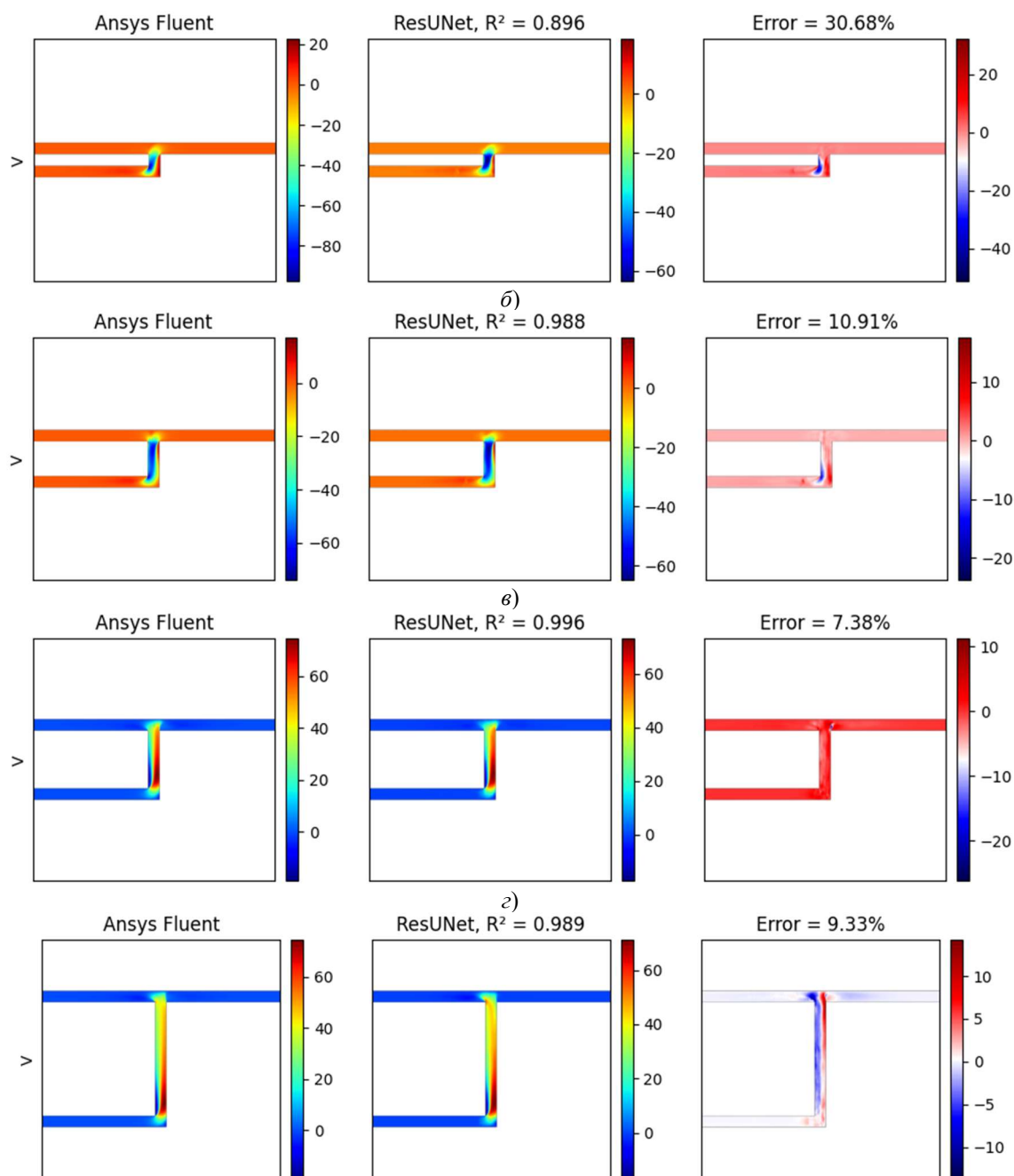


Рис. 12. Результаты аппроксимации поля компоненты скорости V для четырех конфигураций узла с вытяжным тройником: а) $l_b = 1b$ и $r = 0.1/0.9$; б) $l_b = 3b$ и $r = 0.3/0.7$; в) $l_b = 5b$ и $r = 0.7/0.3$; г) $l_b = 10b$ и $r = 0.9/0.1$ (иллюстрация авторов)

Fig. 12. Results of approximation of the field of the velocity component V for four configurations of a node with an exhaust tee: а) $l_b = 1b$ and $r = 0.1/0.9$; б) $l_b = 3b$ and $r = 0.3/0.7$; в) $l_b = 5b$ and $r = 0.7/0.3$; г) $l_b = 10b$ and $r = 0.9/0.1$ (illustration by the authors)

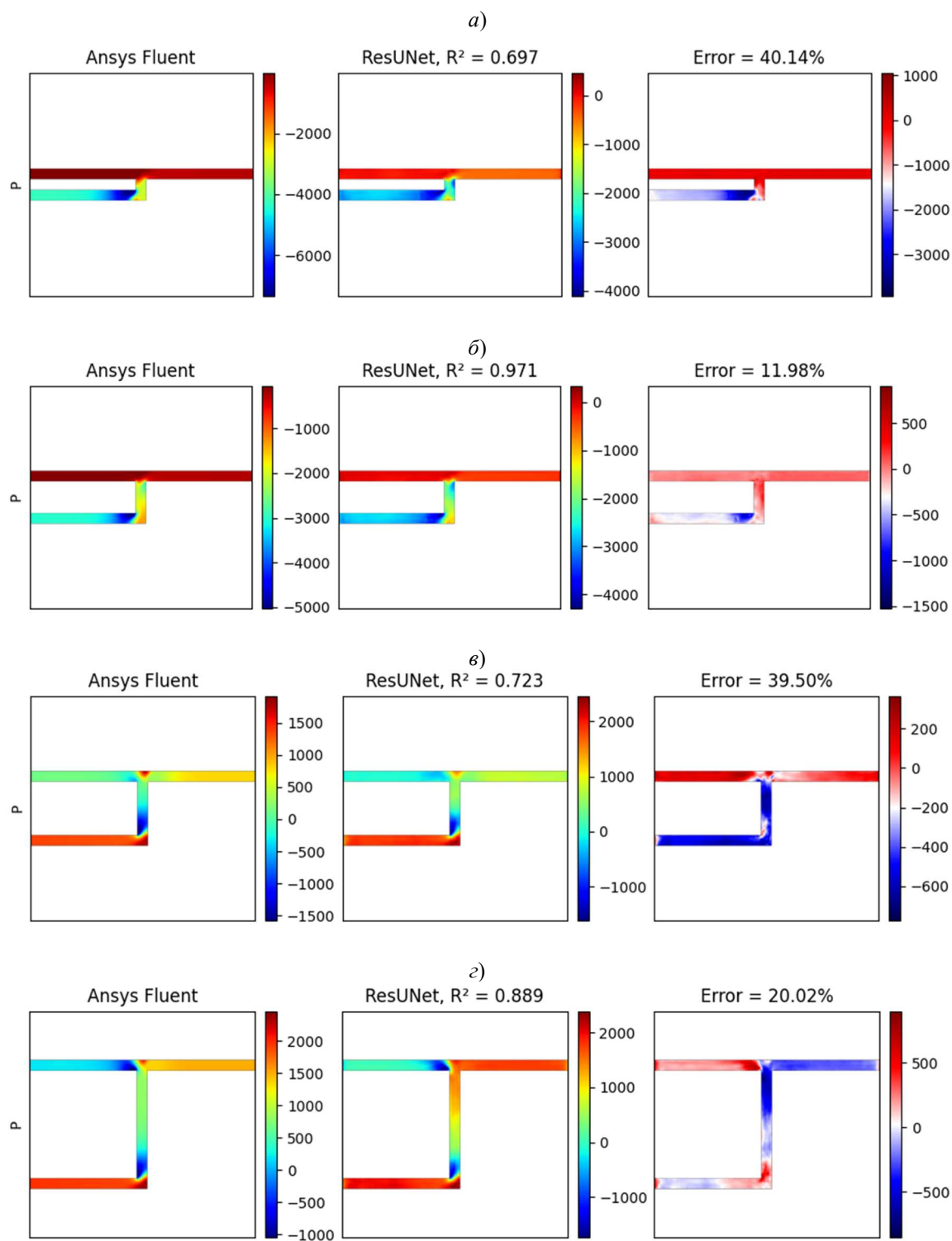


Рис. 13. Результаты аппроксимации поля компоненты скорости P для четырех конфигураций узла с вытяжным тройником: а) $l_b = 1b$ и $r = 0,1/0,9$; б) $l_b = 3b$ и $r = 0,3/0,7$; в) $l_b = 5b$ и $r = 0,7/0,3$; г) $l_b = 10b$ и $r = 0,9/0,1$ (иллюстрация авторов)

Fig. 13. Results of approximation of the field of the velocity component P for four configurations of a node with an exhaust tee: а) $l_b = 1b$ and $r = 0.1/0.9$; б) $l_b = 3b$ and $r = 0.3/0.7$; в) $l_b = 5b$ and $r = 0.7/0.3$; г) $l_b = 10b$ and $r = 0.9/0.1$ (illustration by the authors)

4. Заключение

1. Разработан метод преобразования полей давления и компонент скорости, полученных в вычислительном комплексе Ansys Fluent, в матричный формат с формированием признаков карт (бинарная маска, поля SDF, граничные условия).
2. В результате исследования оптимизирована архитектура стандартной свёрточной сети ResUNet: ядро 7×7 , прогрессия фильтров до 128, вес градиентного штрафа $w_{grad}=0,4$. Это позволило выявлять нелинейные пространственные закономерности течений даже при разреженной обучающей выборке.
3. Достигнута достаточная точность аппроксимации компонент скорости ($R^2 > 0,98$, WAPE 8–12 %), подтверждающая применимость модели для предсказания зон отрыва потока при допустимой погрешности до 15 %.
4. Показана несколько более высокая погрешность при прогнозировании поля давления ($R^2 \approx 0,88$, WAPE $\approx 27,5$ %). Это объясняется повышенной чувствительностью скалярных полей к градиентным штрафам и ограничениями единого декодера при совместной обработке векторных и скалярных величин.

Перспективным направлением развития этого метода является более тонкая настройка архитектуры свёрточной сети с отдельными декодерами для скорости и давления, снижение скорости обучения на поздних эпохах или выбор другой функции активации.

Список литературы/References

1. Зиганшин А.М., Логачев К.И. Вихревая вентиляция. Профилированные элементы систем вентиляции сниженной энергоёмкости. М.- Ижевск: Институт компьютерных исследований, 2022. 336 с.
Ziganshin A.M., Logachev K.I. Vortex ventilation. Profiled elements of ventilation systems with reduced energy consumption. Moscow; Izhevsk: Institute of Computer Research, 2022. 336 p.
2. Идельчик И.Е. Справочник по гидравлическим сопротивлениям. М.: Машиностроение, 1992. 672 с.
Idelchik I.E. Handbook of hydraulic resistance. Moscow: Mechanical Engineering, 1992. 672 p.
3. Зиганшин А.М., Озеров А.О., Солодова Е.Э. Численное исследование течения в П-образном отводе и снижение его сопротивления // Известия ВУЗов. Строительство. 2019. т. 721, № 1. С. 82–93. DOI: 10.32683/0536-1052-2019-721-1-82-93.
Ziganshin A.M., Ozerov A.O., Solodova E.E. Numerical study of flow in a U-shaped outlet and reduction of its resistance. News of Higher Educational Institutions. Construction. 2019. No. 1. P. 82–93. DOI: 10.32683/0536-1052-2019-721-1-82-93.
4. Солодова Е.Э. Особенности численного моделирования течений в Z-образных отводах систем вентиляции и кондиционирования зданий и сооружений // Известия Казанского государственного архитектурно-строительного университета. 2021. т. 55, № 1. С. 71–84. DOI: 10.52409/20731523_2021_1_71.
Solodova E.E. Features of numerical modeling of flows in Z-shaped bends of ventilation and air conditioning systems of buildings and structures. News of KSUAE. 2021; Vol. 55 No. 1. P. 71–84. DOI: 10.52409/20731523_2021_1_71
5. Зиганшин А.М., Фасыхова Г.Р., Логачев К.И., Альмухамедова Д.Б., Булыгина А.А. Компьютерное моделирование течения воздуха в узле «приточный несимметричный тройник - колено» // Известия вузов. Строительство. 2025. № 7. С. 79–92. DOI: 10.32683/0536-1052-2025-799-7-79-92.
Ziganshin A.M., Fasykhova G.R., Logachev K.I., Almukhamedova D.B., Bulygina A.A. Numerical modeling of air flow in the “supply nonsymmetrical tee – sharp elbow” unit. News of Higher Educational Institutions. Construction. 2025; No.7. P. 79–92. DOI: 10.32683/0536-1052-2025-799-7-79-92.
6. Morozova N., Trias F.X., Capdevila R., Pérez-Segarra C.D., Oliva A. On the feasibility of affordable high-fidelity CFD simulations for indoor environment design and control

- // Building and Environment. Elsevier BV, 2020. Vol. 184. P. 107144. DOI: 10.1016/j.buildenv.2020.107144.
7. Plana-Riu J., Trias F.X., Alsalti-Badellou A., Álvarez-Farré X., Colomer G., Oliva A. Exploiting repeated matrix block structures for more efficient CFD on modern supercomputers // Computer Physics Communications. Elsevier BV, 2026. Vol. 326. P. 110232. DOI: 10.1016/j.cpc.2026.110232.
 8. Olas T., Szustak L., Wyrzykowski R., Olas M., Lapegna M. Advances in Adapting Memory-Bound CFD Computations to RISC-V Multicore Architecture // Lecture Notes in Computer Science. Springer Nature Switzerland, 2025. P. 151–166. DOI: 10.1007/978-3-031-97632-2_11.
 9. Guo D., Heaney C.E., Chen B., Tang J., Cammarano A., Kumar P., Pain C.C. Optimal control of ventilation in enclosed spaces using Adversarial Neural Networks // Building and Environment. Elsevier Ltd, 2026. Vol. 287, No. PB. P. 113858. DOI: 10.1016/j.buildenv.2025.113858.
 10. Mao R., Lan Y., Liang L., Yu T., Mu M., Leng W., Long Z. Rapid CFD Prediction Based on Machine Learning Surrogate Model in Built Environment: A Review // Fluids. 2025. Vol. 10, No. 8. P. 1–28. DOI: 10.3390/fluids10080193.
 11. Guo Y., Wang Y., Cao Y., Long Z. A New Optimization Design Method of Multi-Objective Indoor Air Supply Using the Kriging Model and NSGA-II // Applied Sciences (Switzerland). 2023. Vol. 13, No. 18. DOI: 10.3390/app131810465.
 12. Jansson T., Nilsson L., Redhe M. Using surrogate models and response surfaces in structural optimization - With application to crashworthiness design and sheet metal forming // Structural and Multidisciplinary Optimization. 2003. Vol. 25, No. 2. P. 129–140. DOI: 10.1007/s00158-002-0279-y.
 13. Tao J., Sun G. Application of deep learning based multi-fidelity surrogate model to robust aerodynamic design optimization // Aerospace Science and Technology. 2019. Vol. 92. P. 722–737. DOI: 10.1016/j.ast.2019.07.002.
 14. Baitureyeva A., Yang T., Wang H.S. Development of Machine Learning-Aided Rapid CFD Prediction for Optimal Urban Wind Environment Design // Sustainable Cities and Society. 2025. Vol. 121. P. 106208. DOI: 10.1016/j.scs.2025.106208.
 15. Wang T., Li X., Lu Y., Dong L., Shi F., Lin Z. An efficient thermal comfort prediction method for indoor airflow environment using a CFD-based deep learning model // Building and Environment. 2025. Vol. 267. P. 112246. DOI: 10.1016/j.buildenv.2024.112246.
 16. Vandewiel M.R., Eneyew D.D., Awol A.D., Capretz M.A.M., Bitsuamlak G.T. Approximating CFD simulations of natural ventilation: A deep surrogate model with spatial attention mechanism // Journal of Building Engineering. 2025. Vol. 105. P. 112425. DOI: 10.1016/j.job.2025.112425.
 17. Tian X., Cheng Y., Lin Z. Modelling indoor environment indicators using artificial neural network in the stratified environments // Building and Environment. 2022. Vol. 208. P. 108581. DOI: 10.1016/j.buildenv.2021.108581.
 18. Liang L., Zhao M., Wang Y., Long Z., Yin H. Optimization of indoor thermal environment for high-altitude sentry buildings with attached ventilation based on proper orthogonal decomposition // Building and Environment. 2025. Vol. 267. DOI: 10.1016/j.buildenv.2024.112200.
 19. Wei C., Ooka R. Indoor airflow field reconstruction using physics-informed neural network // Building and Environment. 2023. Vol. 242. DOI: 10.1016/j.buildenv.2023.110563.
 20. Raissi M., Perdikaris P., Karniadakis G.E. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations // Journal of Computational Physics. 2019. Vol. 378. P. 686–707. DOI: 10.1016/j.jcp.2018.10.045.
 21. Shao X., Liu Z., Zhang S., Zhao Z., Hu C. PIGNN-CFD: A physics-informed graph neural network for rapid predicting urban wind field defined on unstructured mesh // Building and Environment. 2023. Vol. 232. DOI: 10.1016/j.buildenv.2023.110056.
 22. XU R., LYU H., YU J., BAO C., WANG H., LIU Y., LIU X. Implicit geometry neural

- network for mesh generation // Chinese Journal of Aeronautics. 2025. Vol. 38, No. 4. DOI: 10.1016/j.cja.2024.11.025.
23. Peng J., Chen X., Liu J. 3DMeshNet: A three-dimensional differential neural network for structured mesh generation // Graphical Models. 2025. Vol. 139. DOI: 10.1016/j.gmod.2025.101257.
 24. Sousa P., Afonso A., Veiga Rodrigues C. Application of machine learning to model the pressure poisson equation for fluid flow on generic geometries // Neural Computing and Applications. 2024. Vol. 36, No. 26. P. 16581–16606. DOI: 10.1007/s00521-024-09935-0.
 25. McConkey R., Yee E., Lien F.S. Deep structured neural networks for turbulence closure modeling // Physics of Fluids. 2022. Vol. 34, No. 3. DOI: 10.1063/5.0083074.
 26. Ling J., Kurzwski A., Templeton J. Reynolds averaged turbulence modelling using deep neural networks with embedded invariance // Journal of Fluid Mechanics. 2016. Vol. 807. P. 155–166. DOI: 10.1017/jfm.2016.615.
 27. Ströfer C.M., Wu J.L., Xiao H., Paterson E. Data-driven, physics-based feature extraction from fluid flow fields using convolutional neural networks // Communications in Computational Physics. 2019. Vol. 25, No. 3. P. 625–650. DOI: 10.4208/CICP.OA-2018-0035.
 28. Kashir B., Ragone M., Ramasubramanian A., Yurkiv V., Mashayek F. Application of fully convolutional neural networks for feature extraction in fluid flow // Journal of Visualization. 2021. Vol. 24, No. 4. P. 771–785. DOI: 10.1007/s12650-020-00732-0.
 29. Yousif M.Z., Yu L., Lim H.C. Physics-guided deep learning for generating turbulent inflow conditions // Journal of Fluid Mechanics. 2022. Vol. 936. DOI: 10.1017/jfm.2022.61.
 30. Buzzicotti M. Data reconstruction for complex flows using AI: Recent progress, obstacles, and perspectives // EPL. 2023. Vol. 142, No. 2. DOI: 10.1209/0295-5075/acc88c.
 31. Зиганшин А.М., Шакирзянов В.И. Применение сверточных нейронных сетей и вычислительной гидродинамики при решении задачи об обтекании препятствия в канале // Вестник Приволжского отделения РААСН: сборник научных трудов. 2025. Т. 28. С. 192–203.
Ziganshin A.M., Shakirzyanov V.I. Application of convolutional neural networks and computational fluid dynamics in solving the problem of flow around an obstacle in a channel // Bulletin of the Volga Branch of the Russian Academy of Agricultural Sciences: collection of scientific papers. 2025. Vol. 28. P. 192–203.
 32. Li Y., Yin P.-K., Chen F.-B., Liang D.-F. Fast and accurate prediction of flow fields around common high-rise buildings under various wind directions using a Fourier skip connection residual U-shaped convolutional neural network // Physics of Fluids. AIP Publishing, 2025. Vol. 37, No. 12. P. 125155. DOI: 10.1063/5.0306780.
 33. Zou X., Tong S., Peng W., Huang Q., Wang J. Residual U-Net for accurate and efficient prediction of hemodynamics in two-dimensional asymmetric stenosis // Physics of Fluids. AIP Publishing, 2025. Vol. 37, No. 7. P. 71925. DOI: 10.1063/5.0274672.
 34. Sa J., Jeon B., Seo Y., Lim J., Choi S.Q. Physics-informed neural networks for complex fluids: opportunities and limitations // Korea-Australia Rheology Journal. Springer Science and Business Media LLC, 2025. Vol. 37, No. 3–4. P. 469–492. DOI: 10.1007/s13367-025-00140-6.
 35. Jacob S.J., Mrosek M., Othmer C., Köstler H. Benchmarking convolutional neural network and graph neural network based surrogate models on a real-world car external aerodynamics dataset // Computers and Fluids. 2025. Vol. 300. DOI: 10.1016/j.compfluid.2025.106760.
 36. Guo X., Li W., Iorio F. Convolutional Neural Networks for Steady Flow Approximation // Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2016. P. 481–490. DOI: 10.1145/2939672.2939738.
 37. Yu J., Lu L., Meng X., Karniadakis G.E. Gradient-enhanced physics-informed neural networks for forward and inverse PDE problems // Computer Methods in Applied

Mechanics and Engineering. Elsevier BV, 2022. Vol. 393. P. 114823. DOI: 10.1016/j.cma.2022.114823.

Информация об авторах

Зиганшин Арслан Маликович, доктор технических наук, доцент, Казанский государственный архитектурно-строительный университет, г. Казань, Российская Федерация

Email: amziganshin@kgasu.ru, ORCID: 0000-0001-7335-7797

Шакирзянов Вадим Ильфатович, инженер-проектировщик ОВиК, ООО «Испытательный центр «Стройэксперт»», г. Казань, Российская Федерация

Email: vadimshakirzyanov@yandex.ru, ORCID: 0009-0002-3045-5797

Information about the authors

Arslan M. Ziganshin, doctor of technical sciences, associate professor, Kazan State University of Architecture and Engineering, Kazan, Russian Federation

Email: amziganshin@kgasu.ru, ORCID: 0000-0001-7335-7797

Vadim I. Shakirzyanov, HVAC design engineer, Stroyexpert Testing Center, Kazan, Russian Federation

Email: vadimshakirzyanov@yandex.ru, ORCID: 0009-0002-3045-5797

Дата поступления: 10.06.2026

Дата принятия: 28.09.2026