

УДК 004.932.2:004.8:004.272.26

РЕСУРСНО-АДАПТИВНАЯ АРХИТЕКТУРА ОКОННОГО ТРАНСФОРМЕРНОГО ВЫВОДА ДЛЯ ПОДГОТОВКИ МРТ-ДАННЫХ К АДДИТИВНОМУ МОДЕЛИРОВАНИЮ

Майтак Р. В., аспирант гр. МРа-251, II курс,
Научный руководитель: протодияконов А. В., к. т. н., доцент
Кузбасский государственный технический университет имени Т.Ф. Горбачева
г. Кемерово

Преобразование магнитно-резонансных данных в геометрию персонализированного изделия начинается с многоклассовой сегментации объёмной сцены. Для автоматизированной технологической подготовки этот этап должен выдавать не только дискретную маску, но и пространственно согласованные поля принадлежности, расстояния до границы и неопределённости. Они используются при топологической проверке, построении полигональной поверхности и назначении локальных технологических допусков. Следовательно, вычислительная архитектура сегментационного модуля должна обеспечивать сохранение физической геометрии, полный охват объёма и предсказуемое потребление ресурсов; аварийное завершение из-за нехватки памяти недопустимо уже потому, что нарушает воспроизводимость всего цифрового маршрута.

Визуальный трансформер представляет изображение последовательно токенов и моделирует зависимости между ними механизмом внимания [1]. В объёмной сегментации этот принцип позволяет учитывать взаимное положение удалённых частей структуры; архитектура UNETR использует трансформерный энкодер в трёхмерной схеме энкодер-декодер [2]. Однако число токенов растёт пропорционально объёму, а матрица полного самовнимания – квадратично относительно их числа. Иерархическое оконное внимание уменьшает стоимость за счёт локальных окон и обмена информацией между смещёнными разбиениями [3]; применимость такого подхода к трёхмерным МРТ-данным показана в Swin UNETR [4]. При этом ограничение внимания внутри сети само по себе не решает проблему инференса на объёмах произвольного размера: выходные массивы и перекрывающиеся фрагменты также занимают память.

В рассматриваемой задаче нейросетевая модель является частью автоматизированной системы технологической подготовки, а не изолированным классификатором. Входные МРТ-серии приводятся к единой физической системе координат; номер и ориентация среза определяются по пространственным атрибутам, а не по имени файла. Это согласуется с представлением

геометрии в стандарте DICOM [5]. Однократное пространственное преобразование и сохранение аффинной матрицы исключают повторный ресэмплинг. Подобный принцип воспроизводимых преобразований реализуется в ИТК и SimpleИТК [6]. После нормализации формируется многоканальный тензор, содержащий интенсивности поддерживаемых серий и маску достоверной области.

Предлагаемая архитектура разделяет три уровня управления. На уровне модели иерархический трансформер объединяет локальное оконное и разреженное глобальное внимание. На уровне объёма детерминированный оператор задаёт перекрывающееся покрытие, порядок обработки и весовую сшивку. На уровне вычислительной среды планировщик до запуска оценивает постоянные и временные массивы и выбирает способ размещения накопителей. Такое разделение принципиально: изменение устройства или режима накопления не меняет весов модели, состава классов и правила принятия решения.

Начальный токен n -го трёхмерного фрагмента строится из его интенсивностного описания, физических координат и признаков технического задания:

$$\mathbf{z}_n^0 = E_{\text{img}}(P_n) + E_{\text{phys}}(\mathbf{r}_n, \mathbf{h}) + E_{\text{task}}(\mathbf{p}_c) \quad (1)$$

Здесь E_{img} формирует признаки локального фрагмента P_n ; E_{phys} кодирует физическую координату центра $\mathbf{r}_n$ и шаг решётки $\mathbf{h}$; E_{task} задаёт активные классы и их приоритеты. Физическое кодирование необходимо потому, что одинаковое смещение в индексах при толщине среза 1 и 5 мм имеет различный метрический смысл. Вектор задания не передаёт сети параметры конкретного принтера, но фиксирует состав требуемого результата, чтобы редкий малый класс не исключался из вычисления из-за частотного дисбаланса.

Полный МРТ-объём $N = (N_x, N_y, N_z)$ обрабатывается окнами $P = (P_x, P_y, P_z)$. Для каждой оси задаётся доля перекрытия ω_d и шаг s_d . Множество начальных позиций формируется с обязательным включением крайнего положения $N_d - P_d$:

$$P_d = \{0, s_d, 2s_d, \dots, N_d - P_d\} \cup \{N_d - P_d\}, \quad s_d = P_d(1 - \omega_d) \quad (2)$$

Операция объединения с последним положением устраняет типичный дефект целочисленного шага: правый край объёма не остаётся непокрытым, даже если $N_d - P_d$ не делится на s_d . Перед инференсом проверяется, что каждое положение принадлежит допустимому диапазону и каждый воксел имеет положительный суммарный вес. Порядок обхода фиксирован лексикографически по координатам окна. Число запусков модели определяется произведением числа положений по трём осям и используется в предварительной оценке времени.

Выход модели включает K полей логитов классов, K полей знакового расстояния и одно поле неопределённости. Локальные значения в центральной части окна считаются более надёжными, чем на его границе, поэтому каждому фрагменту сопоставляется трёхмерная весовая функция Ханна с

положительной нижней границей. Глобальное поле логитов собирается нормированным суммированием:

$$Z_k(\mathbf{r}) = \frac{\sum_{p \in \mathcal{P}} w_p(\mathbf{r}) Z_{k,p}(\mathbf{r})}{\sum_{p \in \mathcal{P}} w_p(\mathbf{r}) + \epsilon} \quad (3)$$

В выражении (3) $Z_{k,p}(\mathbf{r})$ – значение k -го выхода окна p в точке $\mathbf{r}$, $w_p(\mathbf{r})$ – вес окна, а ϵ предотвращает деление на ноль. Та же схема применяется к полям расстояния; для неопределённости сохраняется отдельное правило агрегации. Накопление выполняется как минимум в одинарной точности независимо от точности нейросетевого ядра. Поэтому применение FP16 или bfloat16 внутри трансформера не приводит к последовательному накоплению ошибок округления на перекрытиях.

Оптимизация ресурсов основана на разложении пиковой памяти на четыре части: параметры и рабочие активации модели; входной фрагмент; постоянный накопитель полного объёма; вспомогательные буферы. Консервативная оценка записывается в виде

$$M_{\text{peak}} \approx b C_{\text{in}} P_x P_y P_z + \rho_{\text{act}} M_{\text{net}} + b K N_x N_y N_z + M_{\text{aux}} \quad (4)$$

где b – число байт на элемент; C_{in} – число входных каналов; K – число классов; P и N – размеры окна и объёма; ρ_{act} учитывает активации сети; M_{aux} включает весовое поле и служебные буферы. Член $b K N_x N_y N_z$ в (4) следует трактовать как минимальную постоянную часть одного набора выходов. Если одновременно накапливаются поля принадлежности, расстояния и неопределённости, вместо K подставляется фактическое число каналов C_{out} . Именно этот член часто не учитывается при оценке только по размеру окна, хотя на больших объёмах он превышает полезную нагрузку отдельного запуска модели.

До запуска оркестратор сопоставляет оценку (4) с доступной памятью устройства и резервирует запас η , необходимый драйверу и библиотечным операциям. Выбирается один из трёх режимов, приведённых в таблице 1. Предусловием любого режима является одинаковое покрытие по (2), а постуловием – совпадение размерности и положительный знаменатель в (3). Если ни один режим не удовлетворяет бюджету, задача отклоняется до нейросетевого инференса с диагностикой требуемого ресурса.

6.22.4

Таблица 1 – Режимы размещения накопителей оконного вывода

Режим	Размещение и обмен	Условие выбора
А. Резидентный	Модель, окно и полный накопитель находятся в памяти ускорителя	Оценка с запасом не превышает доступную память устройства
В. Гибридный	Модель и окно – на ускорителе; накопитель – в оперативной памяти	Памяти устройства недостаточно, но полный накопитель помещается в ОЗУ
С. Блочный	Завершённые пространственные блоки выгружаются во внешнее хранилище	Недостаточно ОЗУ; допустим дополнительный обмен без изменения результата

Резидентный режим минимизирует обмен и предпочтителен при наличии достаточного ускорителя. В гибридном режиме логиты окна после инференса преобразуются в FP32 и пакетно переносятся в ОЗУ; это освобождает память устройства от массивов, размер которых определяется полным объёмом. Блочный режим использует тот факт, что для каждой области заранее известно множество перекрывающихся её окон. Блок выгружается только после обработки последнего влияющего окна, поэтому итог (3) совпадает с результатом полного накопителя при том же порядке сложения. Режим не выбирается по факту возникновения ошибки Out of memory: решение принимается до запуска и записывается в паспорт вычисления.

Для воспроизводимости дескриптор задачи содержит размеры исходного и канонического объёмов, физический шаг, P , ω , список начальных позиций, весовую функцию, точность накопителя, версию модели и контрольную сумму весов. Нейросетевой граф экспортируется в ONNX с фиксацией набора операторов [7]. Реализация может использовать MONAI как базовую инфраструктуру медицинского глубокого обучения [8], однако контракт оконного оператора не зависит от конкретной библиотеки. Повторный запуск с тем же дескриптором формирует тот же порядок локальных операций; перенос между режимами изменяет размещение буфера, но не математическое правило сшивки.

6.22.5

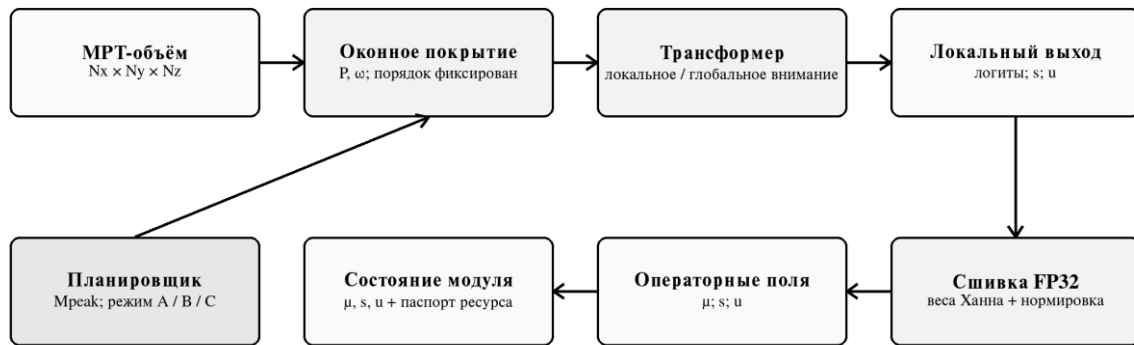


Рисунок 1 – Разделение нейросетевого вывода, сшивки и планирования памяти

Расчётно-алгоритмическая проверка проведена отдельно от оценки точности нейронной сети, чтобы не смешивать ошибки сегментации и свойства инфраструктурного оператора. Исходное значение генератора псевдослучайных чисел равно 23032026. Проверки выполнены в Python 3.12.13 с NumPy 2.3.5 и SciPy 1.17.0 на девяти доступных логических процессорах; полный протокол занял 10,03 с. Контрольные суммы исходного скрипта и файла результатов сохранялись вместе с конфигурацией. Такой эксперимент подтверждает численные свойства покрытия и оценочной модели, но не заменяет испытания на размеченных МРТ-данных.

Для оценки постоянной памяти рассмотрена модель с $K = 8$ классами и тройным выходом: восемь логитов принадлежности, восемь полей знакового расстояния и одно поле неопределённости, всего $\text{Cout} = 17$. Использовалось окно $96 \times 96 \times 96$ и перекрытие 50 %. Полезная нагрузка одного окна, включающая вход, выходные каналы и весовой буфер в одинарной точности, составила 64,1 МиБ без учёта внутренних активаций сети. Результаты расчёта для трёх размеров приведены в таблице 2.

Таблица 2 – Расчёт покрытия и постоянной памяти выходных полей

Размер объёма, воксел	Число окон	Память $\text{Cout} = 17$, МиБ	Локальная нагрузка окна, МиБ
$128 \times 192 \times 192$	18	306	64,1
$160 \times 256 \times 256$	75	680	64,1
$192 \times 320 \times 320$	108	1275	64,1

При $128 \times 192 \times 192$ вокселях по осям формируются 2, 3 и 3 положения, то есть 18 окон. Для среднего объёма получены $3 \times 5 \times 5 = 75$ окон, для наибольшего – $3 \times 6 \times 6 = 108$. Число локальных запусков увеличивается примерно в шесть раз, тогда как полезная нагрузка одного окна остаётся постоянной. Постоянный накопитель, напротив, возрастает с 306 до 1275 МиБ. Поэтому

уменьшение P не гарантирует решения проблемы памяти: оно снижает локальную нагрузку и одновременно увеличивает число окон, но почти не меняет размер полного выхода. Практическое преимущество режимов B и C состоит именно в переносе или разбиении постоянного накопителя.

Корректность весовой сшивки проверялась на синтетическом непрерывном поле $48 \times 64 \times 72$. Поле обрабатывалось 24 окнами 32^3 с перекрытием 50 % и весами Ханна; локальный оператор был тождественным. Максимальное абсолютное отклонение восстановленного поля от эталона составило $2,776 \cdot 10^{-16}$, среднее – $1,491 \cdot 10^{-17}$. Полученный уровень соответствует округлению двойной точности и показывает отсутствие вычислительно различимого шва у самого оператора. Проверка не доказывает, что нейронная модель одинаково предсказывает один объект в разных контекстах окна; она изолирует и подтверждает только нормированное накопление.

Для реального трансформерного вывода дополнительно требуется контролировать контекстную вариабельность. Один и тот же воксел может иметь несколько локальных предсказаний из-за различного положения внутри окна. Весовая схема уменьшает вклад граничных предсказаний, но не устраняет систематическое расхождение. Поэтому вместе с итоговым полем сохраняются дисперсия локальных логитов и карта числа перекрытий. Участки, в которых расхождение превышает установленный на валидационной выборке порог, включаются в поле неопределённости и не должны автоматически переходить в производственную геометрию.

Ресурсная адаптация также не является методом повышения анатомической точности. Она гарантирует вычислительную осуществимость, полноту покрытия и единообразие агрегирования, но качество сегментации должно отдельно оцениваться по покласовым объёмным и поверхностным метрикам [9]. Нельзя делать вывод о пригодности персонализированной модели только из того, что оконный оператор воспроизводим. После сегментации необходимы топологическая верификация, построение замкнутой поверхности и проверка локального бюджета геометрического преобразования.

Связь предложенной архитектуры с технологической подготовкой состоит в формальном постулате сегментационного состояния. В последующий этап передаются поля μk , sk и u , исходное и каноническое пространственные преобразования, а также паспорт вычислительного ресурса. Состояние отклоняется при неполном покрытии, нулевом знаменателе весов, несогласованной размерности каналов, недопустимом преобразовании координат или превышении установленного бюджета. Тем самым вычислительный ресурс становится проверяемым условием выпуска промежуточного артефакта, а не внешней характеристикой рабочей станции.

Для практического внедрения оценку (4) следует уточнять по фактически зарегистрированному пиковому потреблению конкретной модели и устройства. Коэффициент $racst$ зависит от реализации внимания, размера пакета,

контрольных точек и формата чисел; его нельзя переносить между архитектурами без измерения. Аналогично, выбор между режимами В и С должен учитывать пропускную способность шины и хранилища. Однако консервативная оценка остаётся полезной даже при неточных коэффициентах: она отделяет зависящую от окна часть от зависящей от полного объёма и заранее обнаруживает невозможную конфигурацию.

Таким образом, предложена ресурсно-адаптивная архитектура оконного трансформерного вывода, в которой локально-глобальная модель, детерминированное покрытие и планировщик памяти связаны явными контрактами. Обязательное крайнее окно в (2) обеспечивает полный охват объёма; нормированная сшивка (3) формирует единое поле без вычислительных швов; оценка (4) учитывает не только окно и активации, но и постоянный многоканальный выход. Для трёх типовых объёмов число окон составило 18, 75 и 108, а постоянная память 17 полей – 306, 680 и 1275 МиБ. На тождественном синтетическом операторе максимальная ошибка сшивки не превысила $2,776 \cdot 10^{-16}$. Полученные результаты обосновывают использование архитектуры как вычислительного модуля автоматизированной технологической подготовки аддитивного изготовления персонализированных объектов по МРТ-данным.

Список литературы:

1. Dosovitskiy A., Beyer L., Kolesnikov A. et al. An Image Is Worth 16×16 Words: Transformers for Image Recognition at Scale // International Conference on Learning Representations. – 2021. – URL: <https://openreview.net/forum?id=YicbFdNTTy> (дата обращения: 31.08.2026).
2. Hatamizadeh A., Tang Y., Nath V. et al. UNETR: Transformers for 3D Medical Image Segmentation // 2022 IEEE/CVF Winter Conference on Applications of Computer Vision. – Waikoloa, 2022. – P. 574–584. – DOI: 10.1109/WACV51458.2022.00181.
3. Liu Z., Lin Y., Cao Y. et al. Swin Transformer: Hierarchical Vision Transformer Using Shifted Windows // 2021 IEEE/CVF International Conference on Computer Vision. – Montreal, 2021. – P. 10012–10022. – DOI: 10.1109/ICCV48922.2021.00986.
4. Hatamizadeh A., Nath V., Tang Y., Yang D., Roth H. R., Xu D. Swin UNETR: Swin Transformers for Semantic Segmentation of Brain Tumors in MRI Images // Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries. Lecture Notes in Computer Science. – Cham : Springer, 2022. – Vol. 12962. – P. 272–284. – DOI: 10.1007/978-3-031-08999-2_22.
5. Digital Imaging and Communications in Medicine (DICOM). Current Edition 2026c [Электронный ресурс]. – URL: <https://www.dicomstandard.org/current> (дата обращения: 31.08.2026).

6. Lowekamp B. C., Chen D. T., Ibáñez L., Blezek D. The Design of SimpleITK // *Frontiers in Neuroinformatics*. – 2013. – Vol. 7. – Art. 45. – DOI: 10.3389/fninf.2013.00045.
7. Open Neural Network Exchange. ONNX Intermediate Representation Specification [Электронный ресурс]. – URL: <https://onnx.ai/onnx/repo-docs/IR.html> (дата обращения: 31.08.2026).
8. Cardoso M. J., Li W., Brown R. et al. MONAI: An Open-Source Framework for Deep Learning in Healthcare // *arXiv*. – 2022. – arXiv:2211.02701. – DOI: 10.48550/arXiv.2211.02701.
9. Taha A. A., Hanbury A. Metrics for Evaluating 3D Medical Image Segmentation: Analysis, Selection, and Tool // *BMC Medical Imaging*. – 2015. – Vol. 15. – Art. 29. – DOI: 10.1186/s12880-015-0068-x.