

© 2021 г. А.В. ЯМАЕВ (rewin1996@gmail.com)
(МГУ им. М.В. Ломоносова; ООО Смарт Энджинс Сервис, Москва),
М.В. ЧУКАЛИНА, канд. физ.-мат. наук (m.chukalina@smartengines.com)
(Федеральный научно-исследовательский центр
“Кристаллография и фотоника” РАН, Москва;
ООО Смарт Энджинс Сервис, Москва),
Д.П. НИКОЛАЕВ, канд. физ.-мат. наук (d.p.nikolaev@smartengines.com)
(Институт проблем передачи информации РАН, Москва;
ООО Смарт Энджинс Сервис, Москва),
А.В. ШЕШКУС (asheshkus@smartengines.com)
(Федеральный исследовательский центр
“Информатика и управление” РАН, Москва;
ООО Смарт Энджинс Сервис, Москва),
А.И. ЧУЛИЧКОВ, д-р физ.-мат. наук (achulichkov@gmail.com)
(МГУ им. М.В. Ломоносова)

НЕЙРОННАЯ СЕТЬ ДЛЯ ПРЕДВАРИТЕЛЬНОЙ ОБРАБОТКИ ДАННЫХ В КОМПЬЮТЕРНОЙ ТОМОГРАФИИ¹

Предложена легковесная шумоподавляющая фильтрующая нейронная сеть, которая реализует этап фильтрации в алгоритме томографической реконструкции свертки и обратного проецирования (FBP). Приведено обоснование нейросетевой архитектуры, выбранной на основе возможности аппроксимации с достаточной точностью операции рамп-фильтрации. Работоспособность сети продемонстрирована на синтетических данных, которые имитируют томографические проекции, зарегистрированные с малой экспозицией. При генерации синтетических данных учитывались квантовая природа рентгеновского излучения, время экспозиции одного кадра и нелинейный отклик детектора ионизирующего излучения. Время выполнения реконструкции с использованием предложенной сети в 11 раз меньше по сравнению с выбранными для сравнения тяжелыми сетями, при качестве реконструкции по метрике *SSIM* выше 0,9.

Ключевые слова: низкодозовая компьютерная томография, нейронные сети, UNet, быстрые вычисления.

DOI: 10.31857/S0005231021100123

1. Введение

Рентгеновская компьютерная томография (КТ) является широко используемым неразрушающим методом восстановления внутренней структуры объекта. В медицине при использовании КТ стремятся минимизировать получаемую пациентом дозу рентгеновского излучения, но чтобы по результату реконструкции врач мог определить патологии и поставить диагноз.

¹ Работа выполнена при частичной финансовой поддержке Российского фонда фундаментальных исследований (проекты № 19-01-00790 и № 18-29-26017).

Уменьшают получаемую дозу, к примеру, следующими двумя способами: уменьшением числа измеренных изображений [1, 2] и уменьшением времени экспозиции.

Уменьшение времени экспозиции ведет к увеличению шума на зарегистрированных данных. Как следствие, это приводит к увеличению шума и на реконструкции. Для его подавления используют алгоритмы следующих классов: на основе статистического анализа [3], нелинейные фильтры [3], итеративные оптимизационные алгоритмы [4] и нейронные сети. Нейронные сети при зашумленных данных могут выступать в роли операции шумоподавления после реконструкции (постпроцессинг), в роли операции шумоподавления на синограмме (препроцессинг) [5] и в роли полноценного оператора реконструкции [6]. Для решения этих задач используют различные нейросетевые модели, к примеру, сверточные нейронные сети, нейронные сети, работающие в вейвлет пространстве (шумоподавление постпроцессингом), а также множество модификаций [7–10] и нейросети Хопфилда (постпроцессинг [11]), сети, работающие как в пространстве реконструкций, так и в пространстве измеренных данных, генеративные нейронные сети (шумоподавление постпроцессингом [12]).

Несмотря на большое разнообразие различных подходов к решению задачи низкодозовой компьютерной томографии, существует мало представителей быстрых нейросетевых методов. Использование быстрых алгоритмов дает возможность производить реконструкцию во время проведения исследования и, следовательно, останавливать исследование в процессе его проведения в момент достижения достаточного качества реконструкции для понижения полученной пациентом дозы [13].

В статье предложен подход, при котором нейронная сеть выполняет операцию фильтрации данных перед этапом обратного проецирования в алгоритме FBP. Таким образом, нейронная сеть выступает не только в качестве фильтра перед обратным проецированием, но и в качестве шумоподавляющего алгоритма. В статье показано, что фильтр в алгоритме FBP можно заменить одним сверточным слоем сети. Нейронная сеть, выполняющая функцию фильтрации данных, позволяет уменьшить объем вычислений и повысить точность реконструкции. Для обучения нейронной сети использовалась функция потерь в виде суммы по обучающей выборке $\{x_i, y_i\}$

$$(1) \quad Loss = - \sum_i SSIM(y_i, BP(F(x_i, v))),$$

где F – функция нейронной сети, x_i – зашумленная синограмма, подающаяся на вход нейронной сети и полученная из проекций с малыми временами экспозиций, v – параметры и веса нейронной сети, BP – операция обратного проецирования, y_i – фантом, послуживший основой для создания синограммы x_i , $SSIM$ (Structure Similarity) – метрика структурного подобия [14]. Метрика $SSIM$ была выбрана по причине того, что проведенные эксперименты показали значительное увеличение качества результатов работы сети после обучения по этой метрике, чем при обучении по метрике L2. Архитек-

тура предложенной нейронной сети основана на модификации архитектуры UNet [15] и свойствах FBP алгоритма и описана в разделе 4. Таким образом, в настоящей статье предлагается обоснованная архитектура нейронной сети, используется специально подобранная под задачу функция потерь и обсуждаются полученные в ходе работы результаты, которые сравниваются с другими обученными нейросетевыми моделями на синтетических данных.

Обобщая изложенное, выделим основную новизну предлагаемого подхода. Во-первых, использование одномерной нейронной сети приводит к независимости алгоритма от угловой схемы измерения, что аналитически невозможно для обычных двумерных сетей по типу UNet. Во-вторых, использование *SSIM* функции потерь. Использование этой функции потерь повышает метрику PSNR и *SSIM* в финале обучения по сравнению с использованием среднеквадратичной функции потерь.

Статья организована следующим образом: в разделе 2 приводится обоснование минимальных требований к фильтрующей синограмме нейросетевой архитектуре, в разделе 3 описывается процесс генерации синтетических данных, используемых для обучения и проверки нейросетевых моделей, в разделе 4 приводится предлагаемая в этой статье нейросетевая архитектура, в разделе 5 описываются и объясняются результаты сравнительных экспериментов, заключение находится в разделе 6.

2. Нейросетевая аппроксимация рамп-фильтра

Каждое измеренное изображение представляется как результат операции Радона над некоторым неизвестным объемом плотностей объекта при некотором заданном угле. Операция Радона – это интегральная сумма вдоль заданного луча. Для двумерной реконструкции, рассматриваемой в этой статье, координаты луча задаются полярной системой координат (r, ϕ) . И соответственно измеренные изображения являются одномерными и зависят только от координаты r . Такие измеренные изображения называются проекциями. Набор проекций же называется синограммой. Для восстановления неизвестного объема плотностей широко используют алгоритм свертки и обратного проецирования (FBP, Filtered Back Projection). Он состоит из двух шагов: (Filter) фильтрация проекций и (Back Projection) обратное проецирование, представимое в виде транспонированного оператора Радона. FBP алгоритм можно представить в операторном виде

$$(2) \quad FBP(Sin) = BP(FT^{-1}(h(w) * FT(Sin))),$$

где BP – обратное проецирование, Sin – синограмма, $h(w)$ – рамп-фильтр, FT – одномерное фурье преобразование вдоль r . При этом рамп-фильтр имеет вид в частотной области

$$(3) \quad h(w) = \frac{|w|}{W},$$

где w – частота, W – нормировочный коэффициент, обычно принимаемый вид максимальной частоты дискретного преобразования Фурье, примененно-

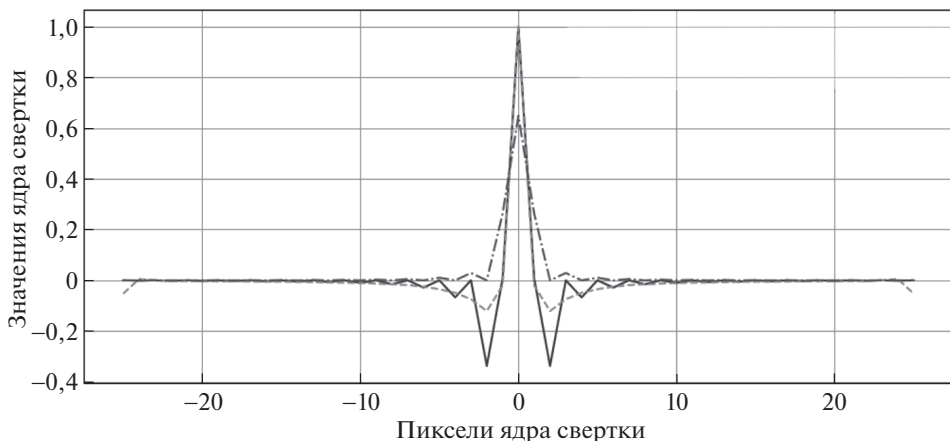


Рис. 1. Ядра свертки рамп-фильтра из приближения и нейросетевого обучения.

го к исходным данным синограммы. Основные особенности этого алгоритма – простота реализации и скорость работы.

Следуя операторному виду FBP , $FT^{-1}(h(w) * FT(Sin))$ представимо в виде операции свертки

$$(4) \quad (H * Sin)(x),$$

где $H(x)$ – представление рамп-фильтра в пространстве синограммы имеет интегральный вид

$$(5) \quad H(x) = \int_{-\infty}^{\infty} h(w)e^{ixw} dw = \int_{-\infty}^{\infty} \frac{|w|e^{ixw}}{W} dw.$$

Поскольку этот интеграл расходящийся, то используют различные его приближения [16]. Одним из них является формула

$$(6) \quad H(x) = W^2 \frac{1 + \cos(Wx)}{\pi^2 - x^2 W^2}.$$

На рис. 1 в соответствии с (6) сплошной линией изображено приближенное ядро свертки, штриховой линией изображено обученное ядро свертки и штрихпунктирной линией – обратное Фурье преобразование рамп-фильтра.

Однако FBP является точным решением только для случая непрерывной незашумленной синограммы [17]. Реальные же данные зашумлены и имеют дискретный вид, что делает нецелесообразным использование FBP для получения высокой точности реконструкций. Однако также следуя изложенному, реконструирующий алгоритм должен иметь возможность с определенно заданной точностью повторить рамп-фильтр. Исследуя свойства стандартных нейросетевых слоев, авторы настоящей статьи пришли к выводу, что рамп-фильтр локализован и с некоторой точностью может быть воспроизведен

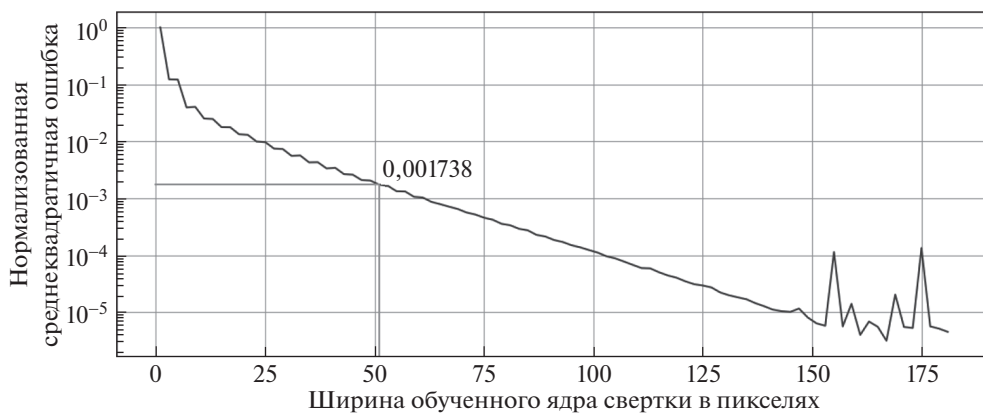


Рис. 2. Нормированный график зависимости среднеквадратичной ошибки обученной линейной одномерной свертки от ширины ядра свертки.

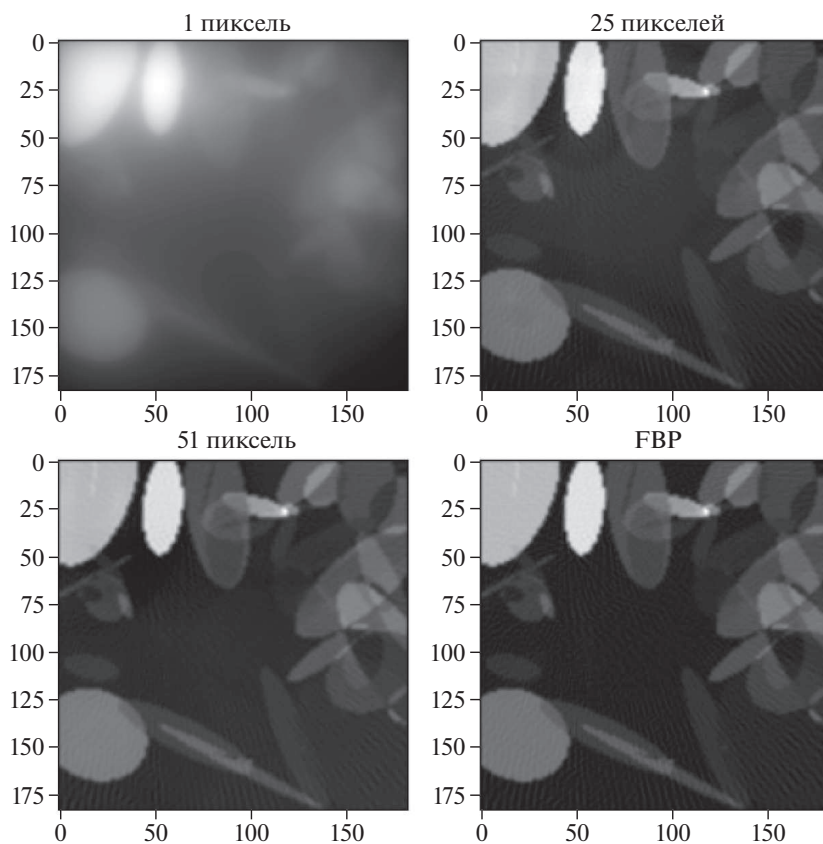


Рис. 3. Реконструкции, полученные при фильтрации синограмм обученными свертками различного размера.

обученной сверткой меньшего размера. Обучая свертки различного размера, от одного пикселя детектора до полного покрытия синограммы, была получена зависимость среднеквадратичной ошибки повторения сверткой рампы-фильтра от ширины ядра используемой свертки. График этой зависимости приведен на рис. 2. Данные для обучения состояли из синограмм и фильтрованных синограмм, построенных от фантомов из раздела 3 и фильтрованных рампы функцией наборов псевдо-дельта функций.

Визуально пропало различие в реконструкциях примерно при свертке размером в 51 пиксель детектора. На основе данных рис. 2 и 3 было решено считать рецептивное поле в 50–51 пиксель проекции достаточным для аппроксимации рампы-фильтра. Весы обученной свертки в выбранной точке показаны на рис. 1. Из этого следует, что нейронные сети со сверточной архитектурой, покрывающей рецептивным полем не менее 51 пикселя проекции, могут быть использованы как фильтрующие нейронные сети.

3. Генерация данных

Для генерации обучающих данных использовался пакет Adler [18]. Пакет позволяет моделировать двумерные изображения наложенных друг на друга эллипсов. Были выбраны только изображения с неотрицательными яркостями.

Синограммы строились с помощью пакетов ODL [18] и Astra Toolbox [19] и зашумлялись на основе квантовых свойств излучения. Синограммы имели ширину в 183 пикселей детектора и 128 углов, распределенных от 0° до 180° градусов (не включая 180°). Зашумленные синограммы использовались как входные данные нейронной сети.

Процедура зашумления происходила по следующей схеме. Выбиралась интенсивность $maxI$ и вычислялась дисперсия аддитивного шума $sigma = maxI/100\,000$. На основе квантовой природы рентгеновского излучения и его пуассоновского поведения рассчитывается зашумленная синограмма по формулам

$$(7) \quad S_e = maxI * e^{-S},$$

где S – незашумленная синограмма,

$$(8) \quad S'_e = P(S_e) + N(0, sigma),$$

где P и N – пуассоновские и нормальные случайные распределения соответственно,

$$(9) \quad I'_0 = P(maxI) + N(0, sigma),$$

$$(10) \quad S' = \log I'_0 - \log S'_e,$$

где S' – синограмма, искаженная случайным шумом. Зашумленная и незашумленная синограммы и их реконструкции показаны на рис. 4. Всего было набрано 20 000 тренировочных синограмм, 5% из которых были выделены под валидацию. Все используемые в статье нейронные сети обучались на одних и тех же тренировочных данных.

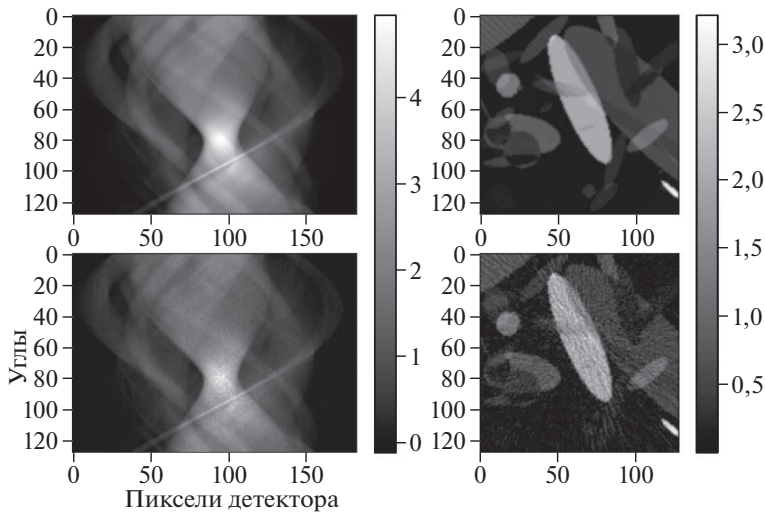


Рис. 4. Примеры данных, используемых в обучении. Слева – незашумленная и зашумленная синограммы, справа – результаты реконструкции.

4. Предлагаемый подход

В медицине требуется быстро обрабатывать данные, что мотивирует использовать одномерные модели вместо двумерных, так как они требуют меньшего объема вычислений при увеличении размерности, чем большинство двумерных моделей. В настоящей статье была разработана одномерная нейронная сеть, принимающая на вход строки синограммы.

Структура модели была основана на UNet-архитектуре, в которой все двумерные свертки заменены на одномерные. При этом количество слоев UNet-архитектуры было подобрано так, чтобы рецептивное поле нейронной сети составляло примерно, но не меньше, 50 пикселей детектора. Как было показано в разделе 2, такого по ширине рецептивного поля достаточно для выполнения операции рамп-фильтрации. Архитектура сети показана на рис. 5. UNet часто применяется для задач сегментации, шумоподавления или преобразования изображения. Более того, для UNet-архитектур характерны высокая обучаемость на небольшом количестве данных и высокая скорость работы. На основе этих характеристик UNet в одномерном виде была выбрана как целевая архитектура.

5. Результаты

Основываясь на том, что рамп-фильтр можно с достаточной точностью аппроксимировать локализованной одномерной сверткой, в статье вместо фильтра в ФВР для реконструкции зашумленных данных применялись легковесные нейронные сети.

В качестве исследуемых моделей были взяты: линейная одномерная свертка (рис. 6), трехслойная сверточная одномерная нейронная сеть (рис. 7), UNet-подобная одномерная архитектура (рис. 5). Эти сети сравниваются по

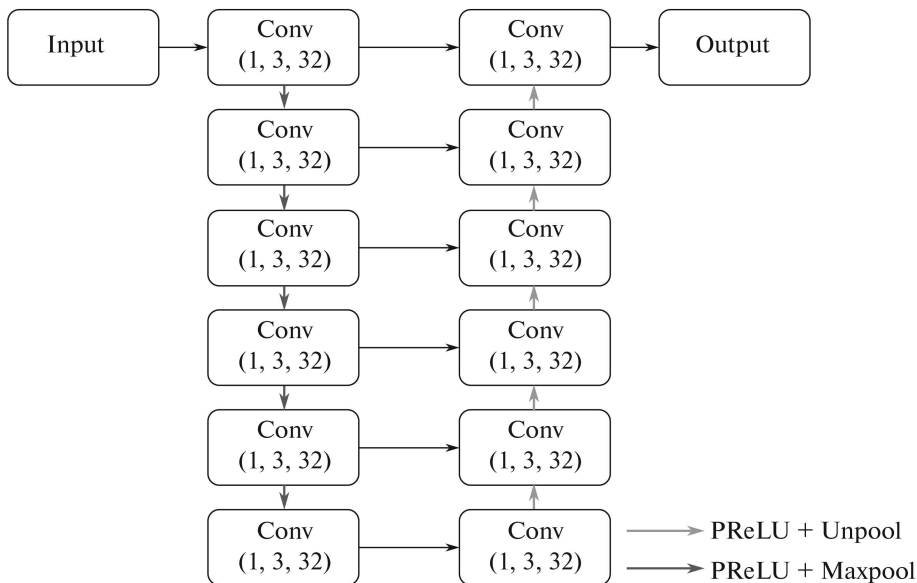


Рис. 5. Изображение архитектуры предложенного одномерного UNet, где в обозначение к Conv первое число – это высота по Y ядра свертки, второе число – это ширина по X ядра свертки, третье число – количество каналов выхода слоя.

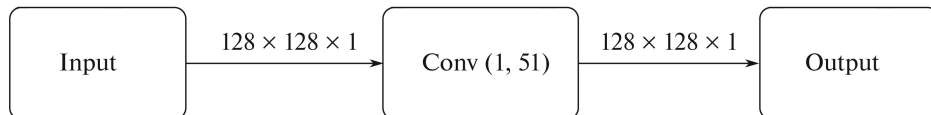


Рис. 6. Изображение архитектуры сети из одной линейной одномерной свертки.

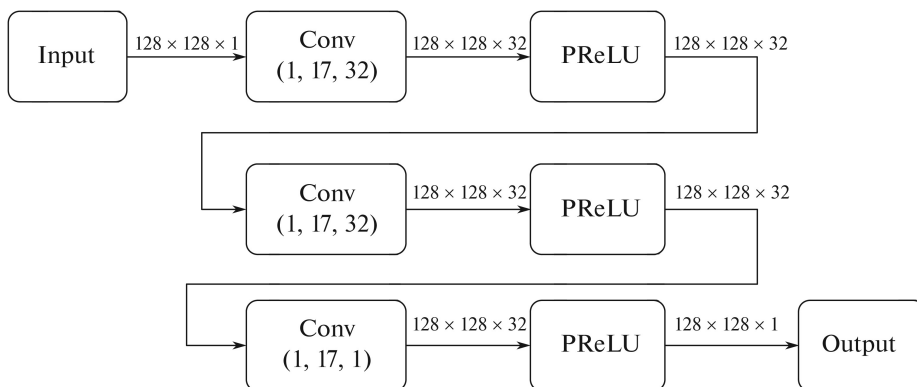


Рис. 7. Изображение архитектуры трехслойной нейронной сверточной одномерной сети.

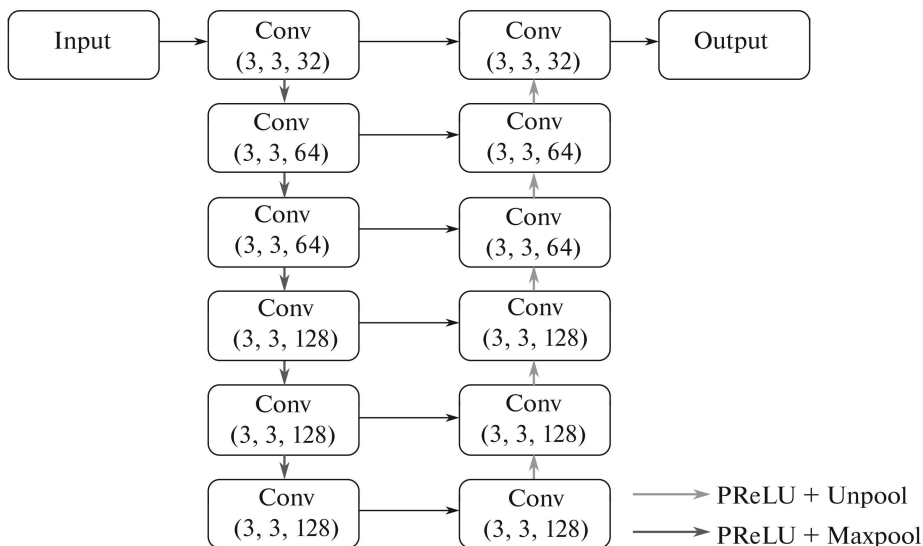


Рис. 8. Изображение архитектуры UNet.

Таблица 1. Таблица с результатами различных архитектур

Нейронная сеть	FBP	Линейная свертка	Трехслойная нейронная сеть	UNet 1D	UNet
Количество умножений на одну синограмму	0	$8,35 \cdot 10^5$	$2,62 \cdot 10^7$	$2,89 \cdot 10^8$	$1,72 \cdot 10^9$
Ширина рецептивная поля	183×128	51	50	50	50×50
Количество обучаемых параметров	0	52	1 633	46 400	1 125 633
Количество умножений на один параметр	0	16 069	16 053	6 237	1 530
PSNR	25	24,5	31,35	33,29	34,03
<i>SSIM</i>	0,5086	0,6143	0,8544	0,9058	0,9223
Время работы, мс	44	21	81	111	100

Таблица 2. Таблица с результатами нейронной сети из LPDR, обученной методом из этой статьи и с полученной в статье LPDR

Нейронная сеть	LPDR	LPDR (<i>SSIM</i>)
Количество умножений на одну синограмму	$4,11 \cdot 10^9$	$4,11 \cdot 10^9$
Ширина рецептивная поля	183×128	183×128
Количество обучаемых параметров	251 980	251 980
Количество умножений на один параметр	16 292	16 292
PSNR	38,73	35,49
<i>SSIM</i>	0,9635	0,9581
Время работы, мс	1214	1214

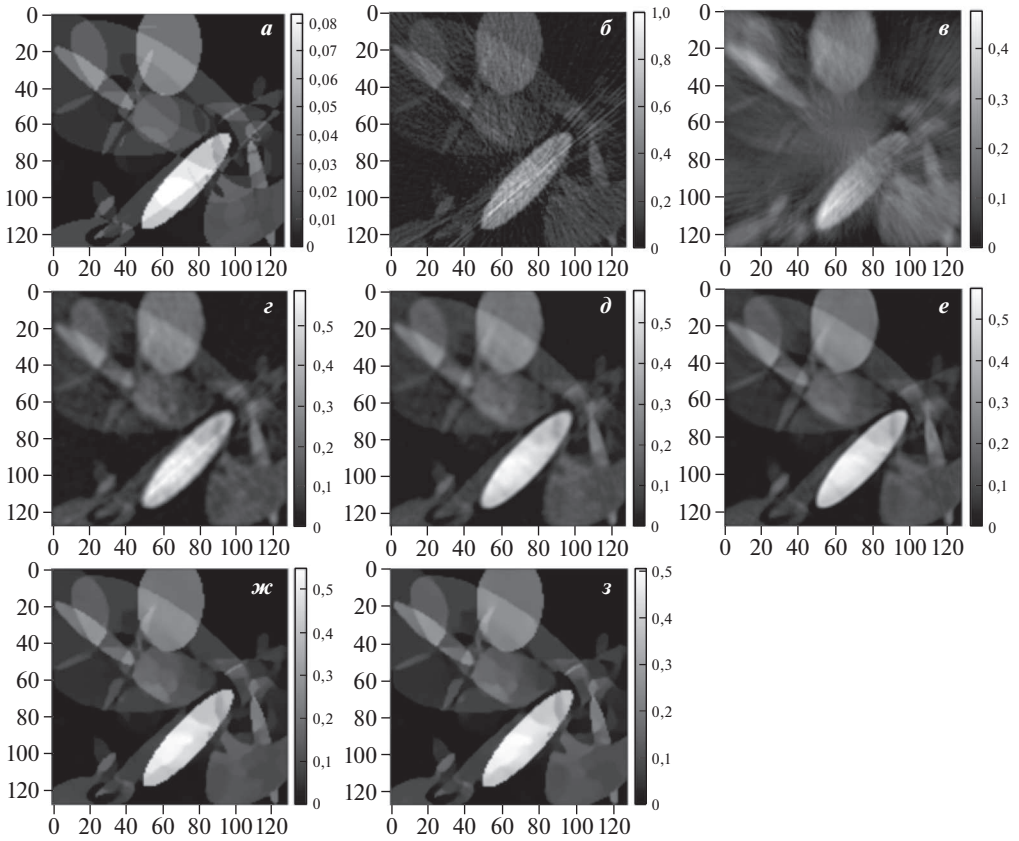


Рис. 9. Результаты различных реконструкций: *а* – фантом, *б* – FBP реконструкция, *в* – линейная свертка, *г* – трехслойная сеть, *д* – UNet1D, *е* – UNet, *ж* – LPDR, *з* – LPDR (*SSIM* функция потерь).

эффективности с двумерными сетями: UNet (рис. 8) и нейросетевой моделью из статьи Learned Primal Dual Reconstruction (LPDR) [6]. Результаты обучения нейронных сетей, а также измеренные характеристики выбранных архитектур представлены в табл. 1 и 2. Результаты реконструкций с помощью сравниваемых архитектур показаны на рис. 9.

При этом размеры свертков в трехслойной нейронной сети, одномерной UNet, двумерной UNet были подобраны так, чтобы рецептивные поля сетей составляли не менее 50 пикселей детектора в ширину, так как эта ширина достаточна для выполнения операции рамп-фильтрации. Также трехслойная нейронная сеть добавлена в сравнение как один из простейших возможных сверточных нейросетевых алгоритмов.

Согласно данным табл. 1 двумерная UNet является лучшей по метрикам PSNR и *SSIM*. Однако ее преимущества перед более простой UNet1D незначительны. Несмотря на то что использование *SSIM* в функции потерь вместо L2 улучшило PSNR и *SSIM* метрики для всех сетей из табл. 1, в экспериментах использование этой функции потерь не улучшило результаты для LPDR

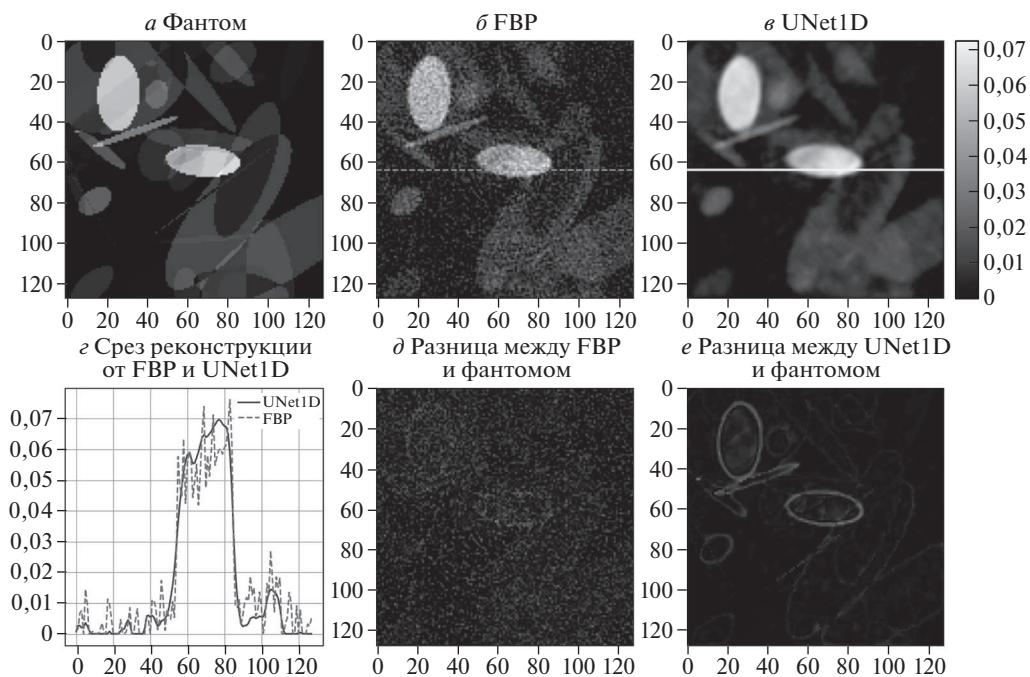


Рис. 10. Анализ ошибки предложенной UNet1D модели: *a* – фантом для проверки, *b* – FBP реконструкция, *c* – реконструкция предложенным в статье подходом, *z* – срезы реконструкций *b* и *c*, отмеченные штриховой и сплошной линиями соответственно, *d* – абсолютная разница между FBP реконструкцией и фантомом, *e* – абсолютная разница между реконструкцией предложенным в статье методом и фантомом. Все полутоновые картинки выполнены в одной шкале значений, указанной справа.

сети, как видно из табл. 2. Время работы нейронных сетей считалось исключительно на процессоре без использования видеокарты, включая требуемые для реконструкции операции преобразования Радона.

На рис. 9 видно, что при использовании UNet1D отсутствует однородный высокочастотный шум, однако границы объектов имеют однопиксельное размытие. Такие артефакты размытия, очевидно, отсутствуют в LPDR сети, это связано с отсутствием в UNet1D каких-либо нейронных слоев в пространстве реконструкций, которые бы поправили недостатки метода обратного проецирования.

Нейросеть, код для запуска нейросети и примеры синтетических данных доступны по ссылке

<https://github.com/ayamaev-se/NNPreprocessingTomography>

6. Заключение

В статье предложен метод фильтрации зашумленной синограммы для задачи низкодозовой компьютерной томографии. Основная идея метода заклю-

чается в применении нейронной сети перед выполнением оператора обратного проецирования. Показано, что рамп-фильтр возможно заменить с достаточной точностью на один одномерный сверточный слой нейронной сети, размер рецептивного поля которого равен 51 пикселю одномерного детектора при ширине рецептивного поля в 183 пикселя детектора. Добавление еще нескольких слоев позволяет подавлять шум, возникающий вследствие сокращения времени экспозиции при регистрации одного томографического изображения. Таким образом, построенная в статье одномерная нейронная сеть заменила фильтр в алгоритме FBP, обучаясь на невязке реконструкции с фантомом по метрике $SSIM$ и обогнав оригинальный FBP оператор. Исследование результатов работы сети проведено на синтетических данных и сравнивалось с широко используемыми нейросетевыми архитектурами. Показано, что предложенный подход производит реконструкцию с точностью выше 0,9 по метрике $SSIM$ в среднем. Кроме того, при таком подходе время выполнения реконструкции на CPU в среднем в 11 раз меньше, чем при использовании LPDR. Также показана эффективность использования оперативной памяти вычислительного устройства при использовании описанного одномерного подхода по сравнению с рассматриваемыми двумерными сетями.

СПИСОК ЛИТЕРАТУРЫ

1. *Hu Z., et al.* An Improved Statistical Iterative Algorithm for Sparse-view and Limited-angle CT Image Reconstruction // Scientific reports. 2017. V. 7. No. 1. P. 1–9.
2. *Swensen S.J., et al.* Screening for Lung Cancer with Low-dose Spiral Computed Tomography // Amer. J. Respiratory and Critical Care Medicine. 2002. V. 165. No. 4. P. 508–513.
3. *Thanh D., et al.* A Review on CT And X-ray Images Denoising Methods // Informatica. 2019. V. 43. No. 2.
4. *Johnson C.A., Sofer A.* A Data-parallel Algorithm for Iterative Tomographic Image Reconstruction // Proc. Frontiers' 99. Seventh Sympos. on the Frontiers of Massively Parallel Computation. IEEE, 1999. P. 126–137.
5. *Yang H.K., et al.* Slice-wise Reconstruction for Low-dose Cone-beam CT Using a Deep Residual Convolutional Neural Network // Nuclear Science and Techniques. 2019. V. 30. No. 4. P. 1–9.
6. *Adler J., Oktem O.* Learned Primal-dual Reconstruction // IEEE Trans. Medical Imaging. 2018. V. 37. No. 6. P. 1322–1332.
7. *Mizusawa S., et al.* Computed Tomography Image Reconstruction Using Stacked U-Net // Computerized Medical Imaging and Graphics. 2021. V. 90. P. 101920.
8. *Han Y., Ye J.C.* Framing U-Net via Deep Convolutional Framelets: Application to Sparse-view CT // IEEE Trans. Medical Imaging. 2018. V. 37. No. 6. P. 1418–1429.
9. *Nakai H., et al.* Quantitative and Qualitative Evaluation of Convolutional Neural Networks with a Deeper U-net for Sparse-view Computed Tomography Reconstruction // Academic radiology. 2020. V. 27. No. 4. P. 563–574.
10. *Zhu L., et al.* Metal Artifact Reduction for X-ray Computed Tomography Using U-net in Image Domain // IEEE Access. 2019. V. 7. P. 98743–98754.

11. *Cierniak R.* A 2D Approach to Tomographic Image Reconstruction Using a Hopfield-type Neural Network // Artificial Intelligence in Medicine. 2008. V. 43. No. 2. P. 113–125.
12. *Yang Q., et al.* Low-dose CT Image Denoising Using a Generative Adversarial Network with Wasserstein Distance and Perceptual Loss // IEEE Trans. Medical Imaging. 2018. V. 37. No. 6. P. 1348–1357.
13. *Bulatov K., et al.* Monitored Reconstruction: Computed Tomography as an Anytime Algorithm // IEEE Access. 2020. V. 8. P. 110759–110774.
14. *Hore A., Ziou D.* Image Quality Metrics: PSNR vs. SSIM // 2010 20th international conference on pattern recognition. IEEE, 2010. P. 2366–2369.
15. *Kofler A., et al.* A U-Nets Cascade for Sparse View Computed Tomography // Int. Workshop on Machine Learning for Medical Image Reconstruction. Springer, Cham, 2018. P. 91–99.
16. *Wei Y., Wang G., Hsieh J.* An Intuitive Discussion on the Ideal Ramp Filter in Computed Tomography (I) // Comput. Math. Appl. 2005. V. 49. No. 5–6. P. 731–740.
17. *Kak A.C., Slaney M., Wang G.* Principles of Computerized Tomographic Imaging. 2002.
18. *Adler J., Kohr H., Öktem O.* ODL — a Python Framework for Rapid Prototyping in Inverse Problems. 2017. Code and documentation available online: <https://github.com/odlgroup/odl>.
19. *Van Aarle W., et al.* The ASTRA Toolbox: A Platform for Advanced Algorithm Development in Electron Tomography // Ultramicroscopy. 2015. V. 157. P. 35–47.

Статъя представена к публикации членом редколлегии А.А. Лазаревым.

Поступила в редакцию 24.01.2021

После доработки 01.06.2021

Принята к публикации 30.06.2021