

МЕТОД УЛУЧШЕНИЯ КАЧЕСТВА ЗАПОЛНЕНИЯ ОБЛАСТЕЙ ИЗОБРАЖЕНИЙ ВЫСОКОГО РАЗРЕШЕНИЯ

© 2021 г. А. В. Москаленко^{а,*}, М. В. Ерофеев^{а,**}, Д. С. Ватолин^{а,***}

^а *Московский государственный университет им. М.В. Ломоносова,
Факультет вычислительной математики и кибернетики,
119991 Москва, ГСП-1, Ленинские горы, д. 1, Россия*

**E-mail: andrey.moskalenko@graphics.cs.msu.ru*

***E-mail: merofeev@graphics.cs.msu.ru*

****E-mail: dmitriy@graphics.cs.msu.ru*

Поступила в редакцию 10.10.2020 г.

После доработки 20.10.2020 г.

Принята к публикации 12.01.2021 г.

В работе рассматривается задача заполнения областей изображений. В последние годы эта область стремительно развивалась, новые нейросетевые методы показывают впечатляющие результаты, однако большинство нейросетевых подходов сильно зависят от разрешения, на котором их обучали. Незначительное увеличение разрешения приводит к серьезным артефактам и неудовлетворительному результату заполнения, из-за чего подобные методы не применимы в средствах интерактивной обработки изображений. В этой статье мы представляем метод, позволяющий решить проблему заполнения областей изображений разного разрешения. Мы также описываем способ более качественного восстановления текстурных фрагментов в заполняемой области. Для этого мы предлагаем использовать информацию из соседних пикселей путем сдвига исходного изображения в четырех направлениях. Предлагаемый подход применим к уже существующим методам без необходимости их переобучения.

DOI: 10.31857/S0132347421030079

1. ВВЕДЕНИЕ

Задача заполнения областей изображения нацелена на восстановление некоторой поврежденной или неизвестной области. На входе алгоритм получает поврежденное изображение, а также маску области, где требуется восстановление. На выходе алгоритм выдает восстановленное изображение, заполняя неизвестную область наиболее реалистичным способом.

В последние годы развитие нейросетевых подходов существенно способствовало появлению различных методов решения этой задачи. Однако нейросетевые подходы сильно привязаны к разрешению, на котором их обучали, из-за недостатка рецептивного поля. Большинство моделей имеет размер входа не превышающий 512 пикселей. В результате они не могут обрабатывать изображения произвольной формы, например, в интерактивных инструментах обработки изображений. Когда разрешение входного изображения повышается, у большинства подходов начинают проявляться значительные артефакты. Пример приведен на рис. 1.

В этой статье предлагается метод заполнения областей, не зависящий от разрешения. Он ис-

пользует coarse-to-fine подход, восстанавливая структуру изображения на низком разрешении и текстуру на высоком. Также, для улучшения качества текстурного заполнения, мы предлагаем использовать сдвиги исходного изображения, тем самым искусственно увеличивая рецептивное поле на величину сдвига. Наш подход теоретически применим к любому существующему методу без необходимости переобучения.

2. СВЯЗАННЫЕ ИССЛЕДОВАНИЯ

Решение поставленной задачи возможно классическими подходами, ориентированными на подбор наиболее подходящего блока для неизвестной области [1–4] из исходного изображения и итеративном заполнении. Такие методы хорошо справляются с заполнением текстурных фрагментов, но плохо работают на сложных структурных объектах.

Развитие нейронных сетей привело к появлению различных глубоких методов [5–8]. Такие алгоритмы используют знания, полученные во время обучения из некоторого множества приме-

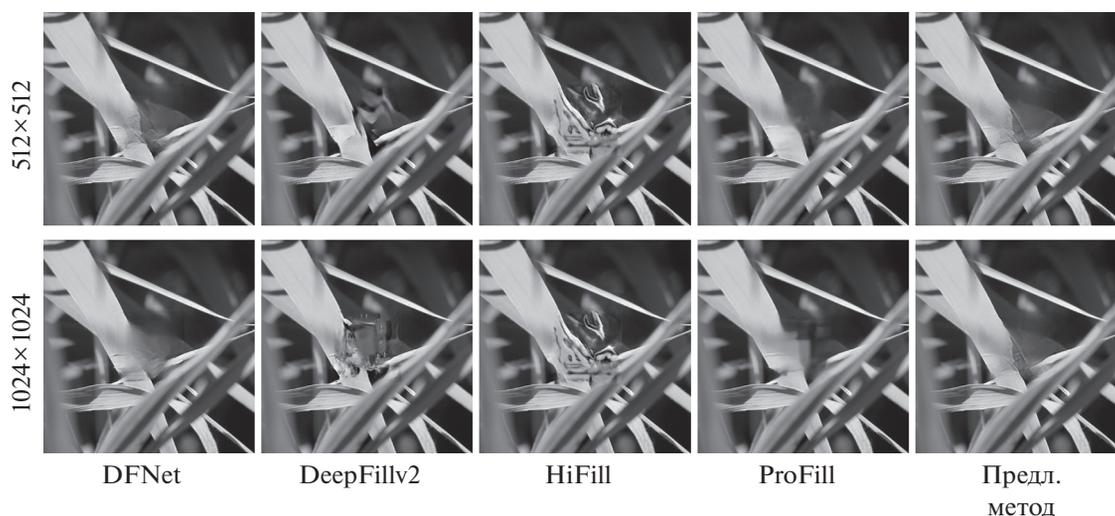


Рис. 1. Пример работы методов при изменении разрешения (маскирован центральный квадрат).

ров. Они лучше классических алгоритмов восстанавливают структурные особенности изображения.

Сложность к обучению нейросетевых подходов добавляет то, что предлагаемые заполнения области могут быть совершенно различными и при этом все быть достаточно реалистичными. Имея исходное изображение становится трудным задать подходящую функцию потерь, наилучшим образом отражающую зрительное восприятие. Как показано в [9], для этой задачи нейросетевые признаки становятся лучшим способом оценки качества заполнения.

Появление генеративно-сопоставительных сетей (GAN) [10] легло в основу создания генеративных методов заполнения изображений [5, 6, 8, 11, 12], которые используют сопоставительную функцию потерь как один из компонентов.

2.1. Модуль контекстного внимания и маскированные свертки

В [11], исследователи предложили заменить классические свертки на маскированные, являющиеся продолжением идей [5]. Также они предложили модуль контекстного внимания, являющегося дифференцируемым аналогом алгоритма выбора наиболее подходящего блока для заполнения неизвестной области. Метод реализован как генеративно-сопоставительная нейросеть. Вместо использования вычислительно сложного модуля контекстного внимания в этой работе мы предлагаем использовать обычные сдвиги исходного изображения как средство заполнения текстуры.

2.2. Глубокая сеть с модулем слияния

Авторы [7] разработали блок слияния, позволяющий нейросети производить альфа-смешивание каждого пикселя в соответствии с предсказанной альфа-картой признаков, полученных на разных разрешениях. Они реализовали архитектуру U-Net [13], используя высокоуровневые нейросетевые признаки из VGG16 [14] в качестве функции потерь. Мы используем предобученную DFNet на первой стадии нашего метода, тем самым восстанавливая структуру неизвестной области на низком разрешении. Мы не применяли блоки слияния в нашей уточняющей сети на втором этапе восстановления текстур, поскольку он изменяет даже немаскированную область.

2.3. Восстановление областей высокого разрешения

В [12] исследователи предложили подход, модифицирующий маскированные свертки, которые требуют значительных вычислительных ресурсов, уменьшив число параметров в нейросети. Также они предложили разделить изображение на высокие и низкие частоты путем вычитания из изображения его версию с использованием фильтра блюра и использования модифицированного блока контекстного внимания для финальной агрегации. Авторы обучали сеть восстановления на небольших, но полноразмерных изображениях. Цель же предлагаемой сети лишь в восстановлении текстурной составляющей, поэтому мы обучали её на небольших случайных фрагментах исходного изображения, используя полноразмерное изображение только при тестировании.

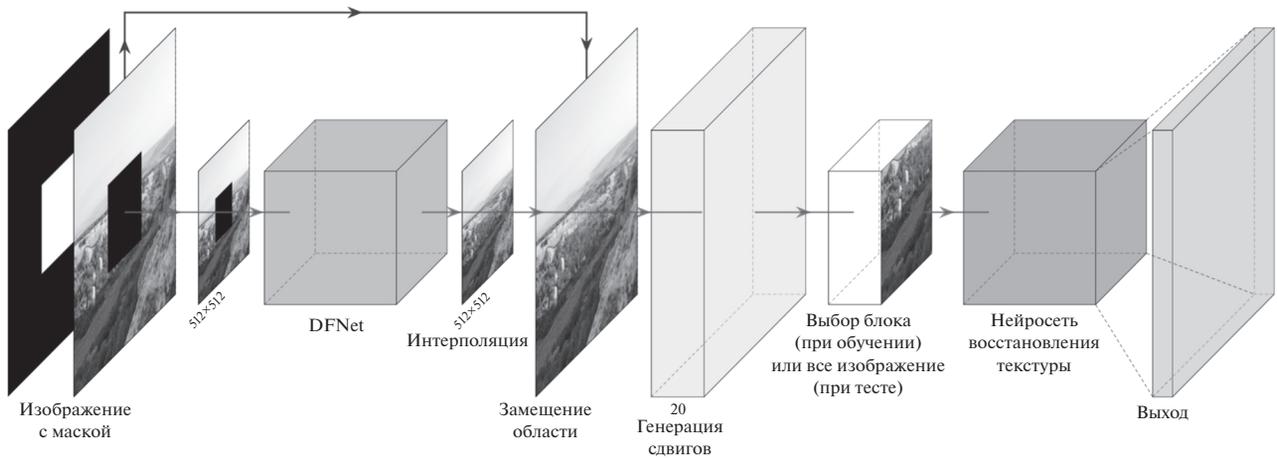


Рис. 2. Общая схема функционирования.

3. ПРЕДЛАГАЕМЫЙ МЕТОД

3.1. Подготовка данных

Для обучения мы взяли все изображения из набора DIV2K [15]. Из каждого изображения мы вырезали случайный наибольший квадрат. Всего в обучающий набор данных вошло 7.218 изображений, в валидационный – 1.650. К каждому изображению мы применили случайную маску из [5]. Для тестирования мы использовали естественные изображения с квадратной маской в центре.

3.2. Общая схема алгоритма

Первая стадия

На первой стадии наш алгоритм восстанавливает структуру изображения на низком разрешении. Сначала мы уменьшаем исходное изображение и маску до 512×512 , затем применяем предобученную DFNet [7] для получения грубого результата заполнения на низком разрешении. Затем мы производим интерполяцию изображения до начального разрешения и выполняем замещение известной области из исходного изображения. Наконец, мы генерируем сдвиги исходного изображения в четырех направлениях: влево, вправо, вниз, вверх. В экспериментах использовались сдвиги на 20% от ширины изображения в пикселях. Отметим, что во время этого процесса мы также пересчитываем маски, помечая все пиксели в областях открытия как невалидные. Таким образом, после первой стадии получается 20-канальное изображение: пять RGB изображений (основное и четыре сдвига) и пять масок.

Вторая стадия

Вторая стадия нацелена на восстановление текстуры. Для предотвращения привязанности сети восстановления к структуре и ускорения процесса

обучения, мы вырезаем случайный 512×512 квадрат из исходного тензора глубины 20 такой, что-маскированная область составляет не менее 10%, но не более 90% от размера этого квадрата.

Отметим, что во время тестирования мы пропускаем стадию выбора патча и применяем сеть для изображения в полном разрешении. Выход сети восстановления является финальным результатом заполнения. Иллюстрация с общей схемой приведена на рис. 2.

3.3. Архитектура сети восстановления текстуры

Предложенная архитектура построена на базе сети U-Net [13] (подробная иллюстрация архитектуры доступна на рис. 3). Отметим, что после каждого слоя, кроме последнего, мы использовали нормализацию из [16]. На иллюстрации отражены размеры сверток кодировщика, размеры всех сверток декодировщика 3×3 .

3.4. Функция потерь

Мы обучали нашу сеть в не генеративно-сопоставительной манере. Аналогично методу предложенному в [7] в качестве функции потерь мы использовали линейную комбинацию:

$$\mathcal{L} = 0.1 \cdot \mathcal{L}_v + 6.0 \cdot \mathcal{L}_1 + 0.1 \cdot \mathcal{L}_p + 240.0 \cdot \mathcal{L}_s, \quad (3.1)$$

где для исходного изображения \mathbf{I} и предсказанного $\hat{\mathbf{I}}$: \mathcal{L}_v – total variation расстояние внутри маскированной области, \mathcal{L}_1 – расстояние, рассчитываемое как

$$\mathcal{L}_1 = \frac{1}{CHW} \|\mathbf{I} - \hat{\mathbf{I}}\|_1, \quad (3.2)$$

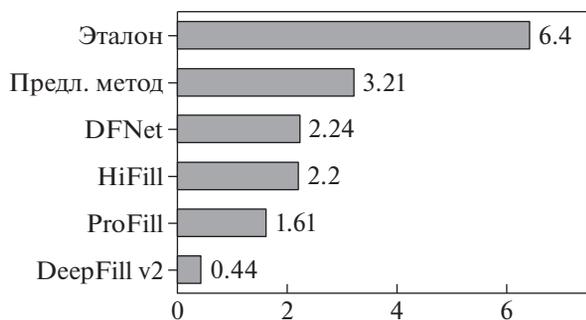


Рис. 5. Результаты субъективного сравнения (разрешение 1024×1024).

и 5. Предложенный алгоритм показал наиболее высокое визуальное качество на обоих разрешениях в сравнении с аналогами.

Объективное сравнение

Из-за сложности получения экспертных оценок мы провели объективные сравнения для других разрешений, хотя это может не быть подтверждено зрительскими оценками. В объективное сравнение мы также добавили результаты метода

Adobe Photoshop 2020, являющегося коммерческим продуктом, реализующим классический метод заполнения. В качестве метрик были выбраны расстояние L1, SSIM [20] и PSNR. Для уменьшения разрешения мы использовали интерполяцию ближайшего соседа. Результаты объективного сравнения представлены в табл. 1. Предложенный алгоритм показал лучшие значения метрик качества, чем аналоги.

5. ЗАКЛЮЧЕНИЕ

В этой работе представлен метод заполнения областей изображений различного разрешения. Мы провели субъективное и объективное сравнения с существующими нейросетевыми подходами, показав, что предложенный метод способен показать более удовлетворительное качество заполнения. Также наш подход применим к уже существующим методам без необходимости их переобучения. Используя [21], предлагаемый метод был встроен в среду интерактивной обработки изображений GIMP. Плагин, код модели, а также полученные в сравнениях изображения доступны в репозитории https://github.com/a-mos/High_Resolution_Image_Inpainting.

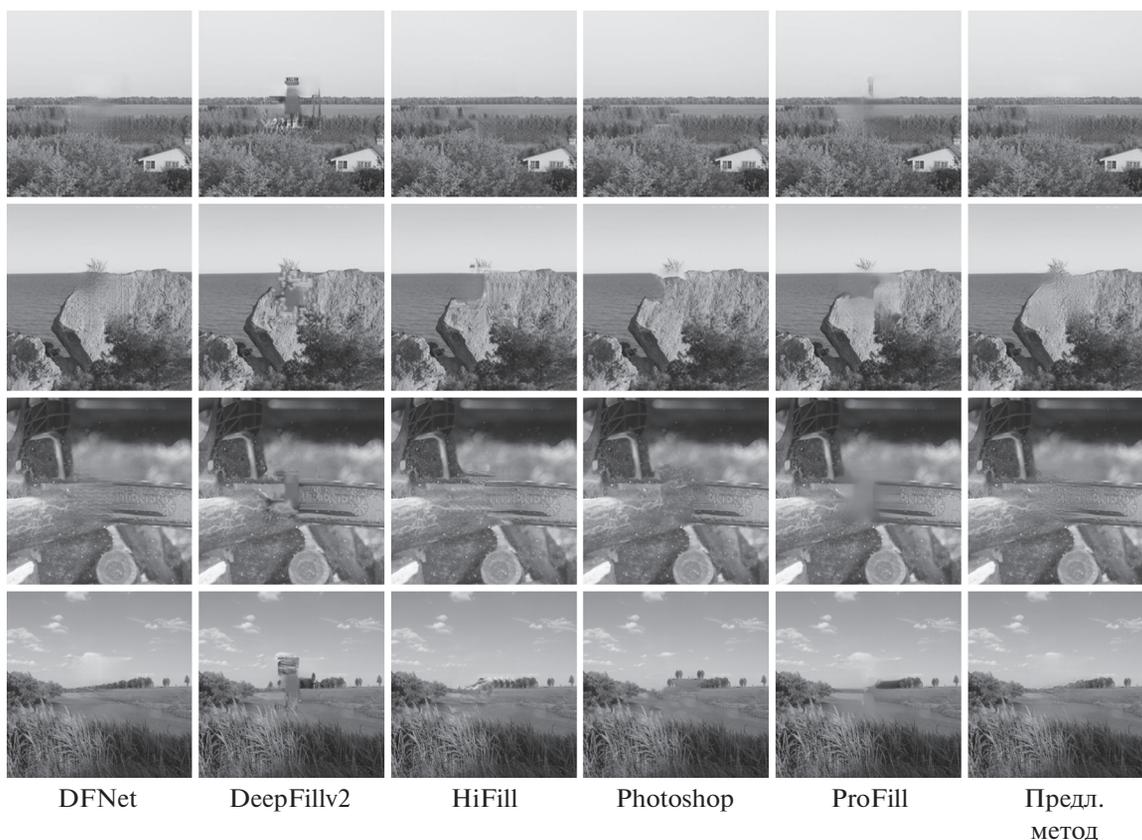


Рис. 6. Примеры работы протестированных методов (1024×1024).

Таблица 1. Результаты объективного сравнения

Модель Метрика	L1	PSNR	SSIM		L1	PSNR	SSIM
DeepFill v2	4.981	22.389	0.938		5.397	21.956	0.944
DFNet	3.592	24.910	0.946		4.132	23.836	0.946
HiFill	4.422	23.667	0.935		4.373	23.738	0.943
Photoshop 2020	4.115	23.789	0.944		13.320	23.756	0.949
ProFill	3.906	24.677	0.950		—	—	—
Ours	3.524	25.175	0.944		3.474	25.311	0.950
Разрешение	1024 × 1024				1536 × 1536		
Модель Метрика	L1	PSNR	SSIM		L1	PSNR	SSIM
DeepFill v2	5.546	21.834	0.946		5.669	21.697	0.948
DFNet	4.345	23.424	0.948		4.633	22.915	0.950
HiFill	4.403	23.696	0.944		4.391	23.710	0.947
Photoshop 2020	4.047	23.863	0.952		4.153	23.659	0.952
Предл. метод	3.447	25.374	0.952		3.434	25.412	0.954
Разрешение	1792 × 1792				2048 × 2048		

6. БЛАГОДАРНОСТИ

Исследования выполнены при финансовой поддержке РФФИ в рамках научного проекта 19-01-00785 а. Для обучения моделей использовался вычислительный кластер IBM Polus факультета вычислительной математики и кибернетики Московского государственного университета имени М.В. Ломоносова.

СПИСОК ЛИТЕРАТУРЫ

1. Drori I., Cohen-Or D., Yeshurun H. Fragment-based image completion // ACM Transactions on Graphics. 2003. V. 22. № 3. P. 303–312.
2. Criminisi A., Perez P., Toyama K. Region filling and object removal by exemplar-based image inpainting // IEEE Transactions on Image Processing. 2004. V. 13. № 9. P. 1200–1212.
3. Barnes C., Shechtman E., Finkelstein A., Goldman D. PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing // ACM Transactions on Graphics (Proc. SIGGRAPH). 2009. V. 28. № 3.
4. Yakubenko A.A., Kononov V.A., Mizin I.S., Konushin V.S., Konushin A.S. Reconstruction of Structure and Texture of City Building Facades // Programming and Computer Software. 2011. V. 37. № 5. P. 260–269.
5. Liu G., Reda F.A., Shih K.J., Wang T.C., Tao A., Catanzaro B. Image Inpainting for Irregular Holes Using Partial Convolutions // The European Conference on Computer Vision (ECCV). 2018. P. 85–100.
6. Yu J., Lin Z., Yang J., Shen X., Lu X., Huang T.S. Generative Image Inpainting with Contextual Attention // IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018. P. 5005–5514.
7. Hong X., Xiong P., Ji R., Fan H. Deep Fusion Network for Image Completion // Proceedings of the 27th ACM International Conference on Multimedia. 2019. P. 2033–2042.
8. Zeng Y., Lin Z., Yang J., Zhang J., Shechtman E., Lu H. High-Resolution Image Inpainting with Iterative Confidence Feedback and Guided Upsampling // arXiv preprint arXiv:2005.11742, 2020.
9. Molodetskikh I., Erofeev M., Vatolin D. Perceptually Motivated Method for Image Inpainting Comparison // CEUR Workshop Proceedings. 2019. V. 2485. P. 131–135.
10. Goodfellow I., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair S., Courville A., Bengio Y. Generative adversarial nets // Advances in neural information processing systems. 2014. V. 27. P. 2672–2680.
11. Yu J., Lin Z., Yang J., Shen X., Lu X., Huang T. Free-Form Image Inpainting With Gated Convolution // Proceedings of the IEEE International Conference on Computer Vision. 2019. P. 4471–4480.
12. Yi Z., Tang Q., Azizi S., Jang D., Xu Z. Contextual Residual Aggregation for Ultra High-Resolution Image Inpainting // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020. P. 7508–7517.
13. Ronneberger O., Fischer P., Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation // International Conference on Medical image computing and computer-assisted intervention. 2015. P. 234–241.
14. Simonyan K., Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition // arXiv preprint arXiv:1409.1556, 2014.
15. Timofte R., Gu S., Wu J., Van Gool L., Zhang L., Yang M.H. NTIRE 2018 Challenge on Single Image Super-Resolution: Methods and Results // The IEEE Conference

- on Computer Vision and Pattern Recognition (CVPR) Workshops. 2018. P. 965.
16. *Ioffe S., Szegedy C.* Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift // arXiv preprint arXiv:1502.03167, 2016.
 17. *Johnson J., Alahi A., Fei-Fei L.* Perceptual Losses for Real-Time Style Transfer and Super-Resolution // European conference on computer vision. 2016. P. 694–711.
 18. *Kingma D.P., Ba J.* Adam: A Method for Stochastic Optimization // arXiv preprint arXiv:1412.6980, 2014.
 19. *Bradley R.A., Terry M.E.* Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons // *Biometrika*. 1952. V. 39. № 3/4. P. 324–345.
 20. *Wang Z., Bovik A., Sheikh H., Simoncelli E.* Image Quality Assessment: From Error Visibility to Structural Similarity // *IEEE transactions on image processing*. 2004. V. 13. № 4. P. 600–612.
 21. *Soman K.* GIMP-ML: Python Plugins for using Computer Vision Models in GIMP // arXiv preprint arXiv:2004.13060, 2020.