УДК 004.85

ИССЛЕДОВАНИЕ И РАЗРАБОТКА СТРАТЕГИИ МАСКИРОВАНИЯ ИЗОБРАЖЕНИЙ ДЛЯ ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ МАСОЧНОГО АВТОЭНКОДЕРА

Килина Мария Леонидовна[1], Буряк Дмитрий Юрьевич[2]

[1]Студент;
Московский государственный университет им. М. В. Ломоносова;
Россия, 119991, г. Москва, ул. Ленинские горы, д. 1;
e-mail: mariya060200@gmail.com.

[2]Кандидат физико-математических наук, доцент;
Московский государственный университет им. М. В. Ломоносова;
Россия, 119991, г. Москва, ул. Ленинские горы, д. 1;
e-mail: dyb04@yandex.ru.

Работа посвящена проблеме повышения эффективности масочного автоэнкодера за счет разработки стратегии маскирования изображений, которая учитывала бы расположение объектов на изображении и позволяла бы скрыть как можно меньше семантически важной информации. В статье представлен обзор существующих методов маскирования изображений, включая стратегии как с учетом, так и без учета структуры изображения. Предложена стратегия наложения масок на основе алгоритма поиска объектов, анализирующего элементарные характеристики фрагментов изображений. Исследование проводится на примере масочного автоэнкодера с ViT в качестве энкодера. Сравнивается эффективность обучения энкодера с использованием предложенной стратегии и с использованием стратегии случайного маскирования изображений.

Ключевые слова: нейронные сети, глубокое обучение, обучение с самоконтролем, моделирование маскированного изображения, модель ViT, масочный автоэнкодер.

Для цитирования:

Килина М. Л., Буряк Д. Ю. Исследование и разработка стратегии маскирования изображений для повышения эффективности масочного автоэнкодера // Системный анализ в науке и образовании: сетевое научное издание. 2025. № 1. С. 8-15. EDN: DBYVHA. URL: https://sanse.ru/index.php/sanse/article/view/647.

RESEARCH AND DEVELOPMENT OF IMAGE MASKING STRATEGY TO IMPROVE MASKED AUTOENCODER EFFICIENCY

Kilina Mariia L.[1], Buryak Dmitriy Yu.[2]

[1]Student;
Lomonosov Moscow State University;
1 Leninskiye Gory, Moscow, 119991, Russia;
e-mail: mariya060200@gmail.com.

[2]PhD in Physical and Mathematical Sciences, associate professor;
Lomonosov Moscow State University;
1 Leninskiye Gory, Moscow, 119991, Russia;
e-mail: dyb04@yandex.ru.

The paper is devoted to the problem of improving the efficiency of masked autoencoder by developing an image masking strategy that considers the object localization in the image and hides as little semantically important information as possible. The article provides an overview of existing methods for masking images, including both considering and not considering the image structure strategies. A masking strategy based on an object detection algorithm that analyzes the elementary characteristics of image fragments is proposed.

The study is carried out on the example of masked autoencoder having ViT as an encoder. The efficiency of training the encoder using the proposed strategy and using the random masking strategy is compared.

Keywords: neural networks, deep learning, self-supervised learning, masked image modeling, ViT model, masked autoencoder.

For citation:
Kilina M. L., Buryak D. Yu. Research and development of image masking strategy to improve masked autoencoder efficiency. *System analysis in science and education*, 2025;(1):8-15 (in Russ). EDN: DBYVHA. Available from: https://sanse.ru/index.php/sanse/article/view/647.

Введение

На сегодняшний день в задачах компьютерного зрения активно используются большие модели, склонные к переобучению, в частности, трансформеры [1], что приводит к необходимости сбора и разметки большого количества данных. Альтернативным способом обучения подобных моделей является моделирование маскированного изображения [2]. При этом подходе модель обучается в качестве энкодера в масочном автоэнкодере, задача которого состоит в том, чтобы реконструировать поданное на вход изображение, частично скрытое маской. Сравнение восстановленного изображения с исходным позволяет вычислить ошибку и провести самоконтролируемое обучение без использования разметки данных, что особенно актуально, когда её получение затруднено.

При этом стратегия наложения масок оказывает влияние на способность модели извлекать семантическую информацию. Случайное наложение масок, как в масочном автоэнкодере *MAE* [3], может скрыть важные элементы изображения и снизить точность модели. А такие методы, как *MST* [4], используют карты внимания от дополнительных моделей, что увеличивает ресурсоемкость обучения.

В качестве альтернативы этим двум методам в работе будет использоваться анализ элементарных характеристик фрагментов изображения, таких как цвет, текстура и т.д. Этот подход позволяет выбрать маску так, чтобы минимизировать сокрытие важных участков, не прибегая к дополнительным моделям.

1. Масочный автоэнкодер

Масочный автоэнкодер [3] – метод, который позволяет провести самоконтролируемое обучение модели при помощи моделирования маскированного изображения. Схема работы метода представлена на рис. 1. Входное изображение разбивается на непересекающиеся патчи, из которых некоторое подмножество скрывается маской. После этого оставшиеся открытыми патчи подаются на вход энкодеру, который вычисляет скрытое представление. Полученное представление подается затем на вход декодеру, который на его основе должен реконструировать изображение. Сравнивая полученное изображение с исходным, модель самоконтролируемо обучается методом обратного распространения ошибки.

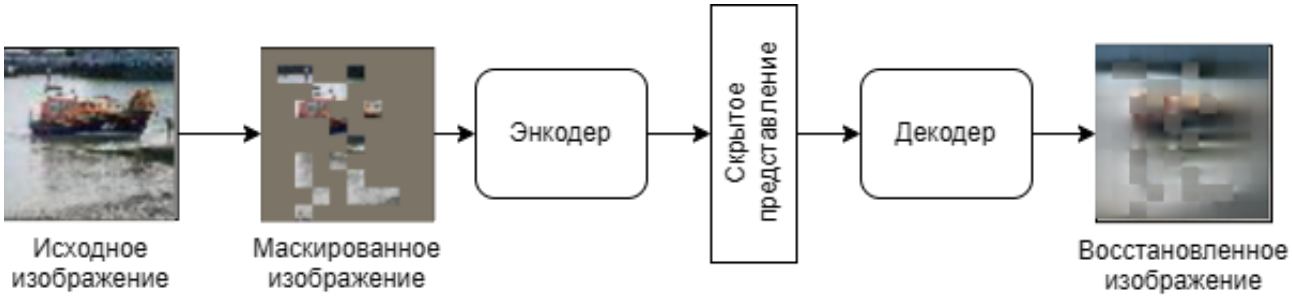


Рис. 1. Масочный автоэнкодер

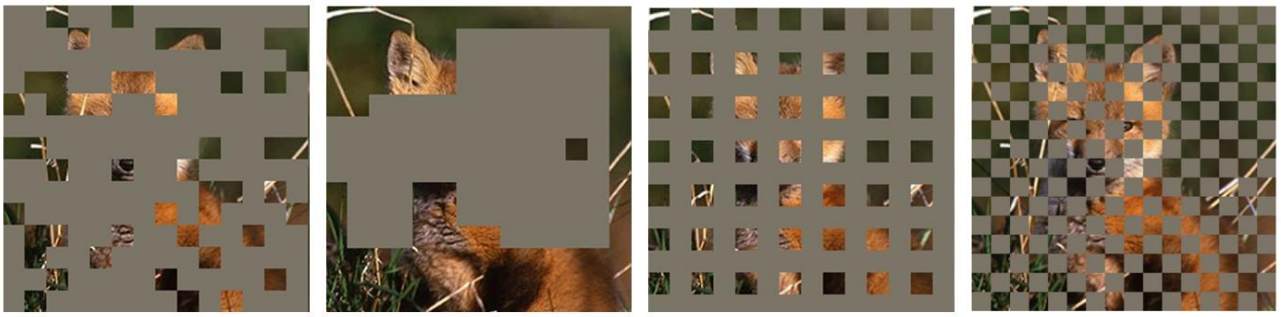
В качестве функции потерь используется среднеквадратичная ошибка (*mean squared error, MSE*) между пикселями восстановленного и исходного изображений. Функция потерь вычисляется только для патчей, которые были скрыты маской.

По завершении самоконтролируемого предобучения, энкодер может быть использован отдельно для решения конкретной поставленной задачи.

2. Существующие методы наложения масок

Стратегия наложения масок существенно влияет на качество обучения модели в масочном автоэнкодере, поскольку от нее зависит, насколько часто на изображении будут скрываться семантически значимые части. Наложение маски подразумевает разбиение изображения на патчи, часть из которых по некоторому принципу выбирается для маскирования. Среди существующих стратегий наложения масок, применяемых в масочных автоэнкодерах, можно выделить стратегии, учитывающие и не учитывающие структуру подаваемого на вход изображения.

Стратегии без учета структуры изображений не используют анализ подаваемого на вход изображения и накладывают маски либо при помощи случайной генерации, либо фиксированным образом. При случайном маскировании [3] (рис. 2, а) набор скрываемых патчей выбирается с помощью генерации массива случайных чисел, из которого выбираются наибольшие значения. Блочное маскирование [2] (рис. 2, б) производится итеративным алгоритмом, на каждой итерации которого к маске добавляется блок патчей, лежащих в случайно выбранном диапазоне. При маскировании решеткой [3] (рис. 2, в) открытым остается каждый четвертый патч в квадрате 2×2. При симметричном маскировании [5] (рис. 2, г) патчи скрываются в шахматном порядке.



а) *б)* *в)* *г)*

Рис. 2. Стратегии без учета структуры изображения

Стратегии, в которых перед наложением маски выделяются важные части изображения, как правило, используют дополнительную предобученную модель-учителя, от которой получают карты внимания. На основе этих карт и выбираются патчи, которые должны быть скрыты. Одним из примеров такого подхода является масочный самоконтролируемый трансформер *MST* [4] (рис. 3). В методе используется предобученный трансформер-учитель, который предоставляет карты внимания для поданного на вход изображения. Маской скрываются патчи, соответствующие участкам со слабым откликом, чтобы избежать сокрытия важных регионов, после чего маскированные данные подаются на вход энкодеру-ученику. Энкодер в свою очередь передает полученное им представление декодеру и классификатору. Для вычисления ошибки используются две составляющие: ошибка реконструкции изображения модели-ученика, а также ошибка между представлениями, полученными на выходе классификаторов учителя и ученика. Таким образом, проводится самоконтролируемое обучение, основанное как на моделировании маскированного изображения, так и на дистилляции знаний.

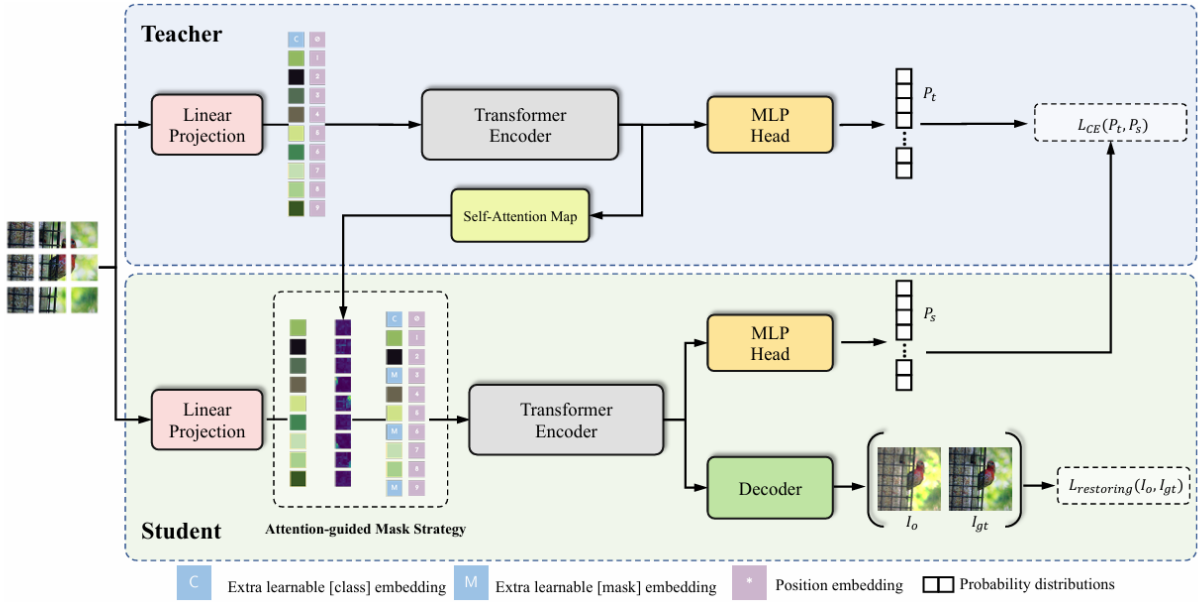


Рис. 3. Масочный самоконтролируемый трансформер MST [4]

3. Стратегия на основе поиска объектов на изображении

Как альтернативу методам, использующим карты внимания, для выбора маски в работе предлагается использовать алгоритм детекции объектов, анализирующий численные характеристики изображений. Одним из таких методов является алгоритм *Selective Search* [6]. В качестве отправной точки алгоритм использует графовый метод сегментации [7], разбивающий изображение на регионы. Далее *Selective Search* объединят регионы друг с другом на основе анализа сходства их цвета, текстуры, размера и формы.

Для анализа сходства регионов на основе цвета алгоритм вычисляет гистограммы из 25 столбцов для трех каналов изображения, получая таким образом 75-мерный дескриптор цвета. Сходство между регионами r_i и r_j вычисляется по формуле:

$$s_{color}(r_i, r_j) = \sum_{k=1}^{n} \min(c_i^k, c_j^k),$$

где c_i^k – значение k-го столбца в цветовом дескрипторе.

Анализ сходства текстуры производится с помощью взятия гауссовских производных по 8 ориентациям для каждого канала изображения. Для каждой из ориентаций и каждого канала строится гистограмма из 10 столбцов и получается 240-мерный дескриптор. Сходство между регионами вычисляется по формуле:

$$s_{texture}(r_i, r_j) = \sum_{k=1}^{n} \min(t_i^k, t_j^k),$$

где t_i^k – значение k-го столбца в дескрипторе.

Сходство по размеру способствует быстрому объединению небольших областей, благодаря чему в разных частях изображения появляются регионы разного масштаба. Сходство регионов по размеру определяется формулой:

$$s_{size}(r_i, r_j) = 1 - \frac{size(r_i) + size(r_j)}{size(im)},$$

где $size(r_i)$ – размер i-го региона, $size(im)$ – размер всего изображения в пикселях.

Сравнимость формы показывает, насколько много пробелов между двумя регионами: если они тесно прилегают друг к другу, то вероятно их стоит объединить. Сходство по форме определяется как:

$$s_{fill} = 1 - \frac{size(BB_{ij}) - size(r_i) - size(r_j)}{size(im)},$$

где BB_{ij} – ограничивающая рамка, охватывающая регионы r_i и r_j.

Общая формула сходства между регионами выглядит следующим образом:

$$s(r_i, r_j) = a_1 s_{color}(r_i, r_j) + a_2 s_{texture}(r_i, r_j) + a_3 s_{size}(r_i, r_j) + a_4 s_{fill}(r_i, r_j),$$

где $a_i \in \{0, 1\}$ – коэффициент, определяющий, используется характеристика или нет.

Выходом алгоритма *Selective Search* является набор ограничивающих рамок, внутри которых предположительно находится объект. Для наложения маски изображение разбивается на патчи, некоторые из которых будут скрыты. Для каждого патча подсчитывается количество рамок, которые его затрагивают. Полученный массив чисел приводится к диапазону $[0,1]$ минимально-максимальной нормализацией:

$$b_i' = \frac{b_i - b_{min}}{b_{max} - b_{min}},$$

где b_i – число рамок, затрагивающих i-й патч.

Чтобы маска не была детерминированной, для каждого патча также генерируется случайное число из диапазона $[0,1]$. После чего вычисляется вес w_i патча по формуле:

$$w_i = \lambda b_i' + (1 - \lambda) n_i,$$

где n_i – сгенерированное случайное число, λ – настраиваемый коэффициент.

Патчи, которым соответствуют наибольшие веса, сохраняются и передаются на вход энкодеру, а заданный процент патчей с наименьшими весами скрывается маской.

4. Подготовка данных для обучения

Для обучения модели был выбран датасет *Tiny ImageNet-200* [8]. Тренировочная выборка этого набора данных состоит из 100000 изображений размера 64×64, разбитых на 200 классов по 500 изображений в каждом классе.

Поскольку *Selective Search* предоставляет детерминированные рамки для одних и тех же изображений, для экономии времени, затрачиваемого на обучение, была проведена предобработка датасета. Для получения рамок ко всей обучающей выборке был применен алгоритм *Selective Search.* Данные о рамках для каждого изображения были сохранены в виде четверок вида *(x,y,w,h)*, где *x* и *y* задают координаты левого верхнего угла рамки, а *w* и *h* – ширина и высота рамки соответственно. Примеры изображений с выделенными на них областями, где предполагается наличие объекта, приведены на рис. 4.

Рис. 4. Примеры выделения рамок

Также в процессе обучения осуществляется аугментация данных. Изображения приводятся к размеру 180×180, случайно обрезаются до размера 160×160, горизонтально отражаются и нормализуются. Эти трансформации были реализованы так, чтобы корректно обрабатывать данные о рамках.

4. Практическая реализация и обучение модели

Все необходимые программные модули реализованы на языке *Python* с использованием фреймворка *Pytorch*. Эксперименты проводились с помощью среды *Jupiter Notebook* с использованием ускорителя *GPU NVIDIA GeForce 3070ti*.

Для обучения в качестве энкодера была выбрана модель *ViT-16B* [9], насчитывающая 12 слоев. В качестве декодера была взята также архитектура трансформера, однако он имеет меньшую глубину по сравнению с энкодером и насчитывает 8 слоев.

Маской скрывалось 80% изображения. Выбор высокого процента сокрытия связан с исследованием *MAE* [3], в котором наилучшие результаты были получены для большой площади маски. Пример работы автоэнкодера со стратегией наложения масок, использующей обнаружение объектов методом *Selective Search* приведен на рис. 5. На рис. 6 проиллюстрирована работа автоэнкодера со случайной маской. На обоих рисунках слева исходное изображение, по центру – изображение после наложения маски, справа – реконструкция, полученная от декодера. Как можно видеть, в первом варианте открытые патчи более сконцентрированы в районе объекта на изображении, в то время как во втором варианте они равномерно разбросаны по изображению.

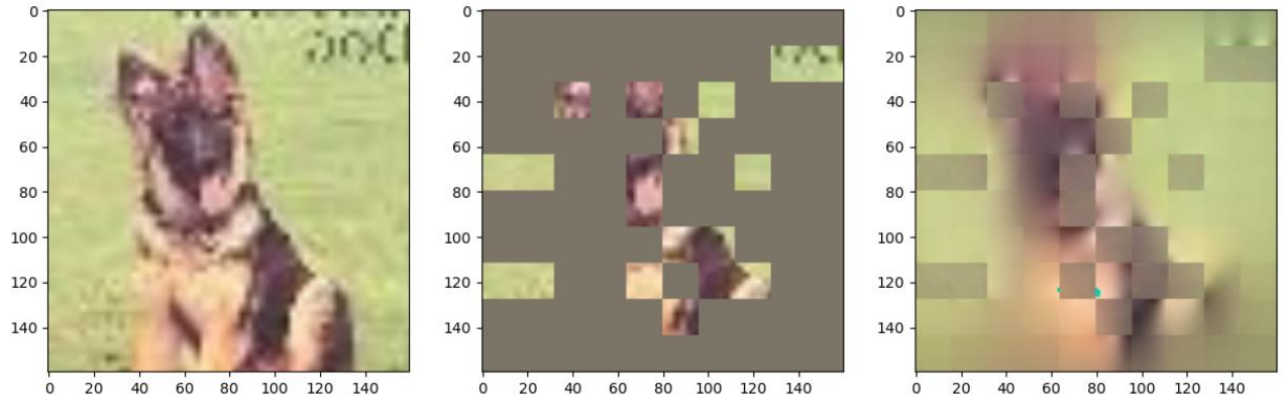


Рис. 5. Работа автоэнкодера с маской, учитывающей положение объектов

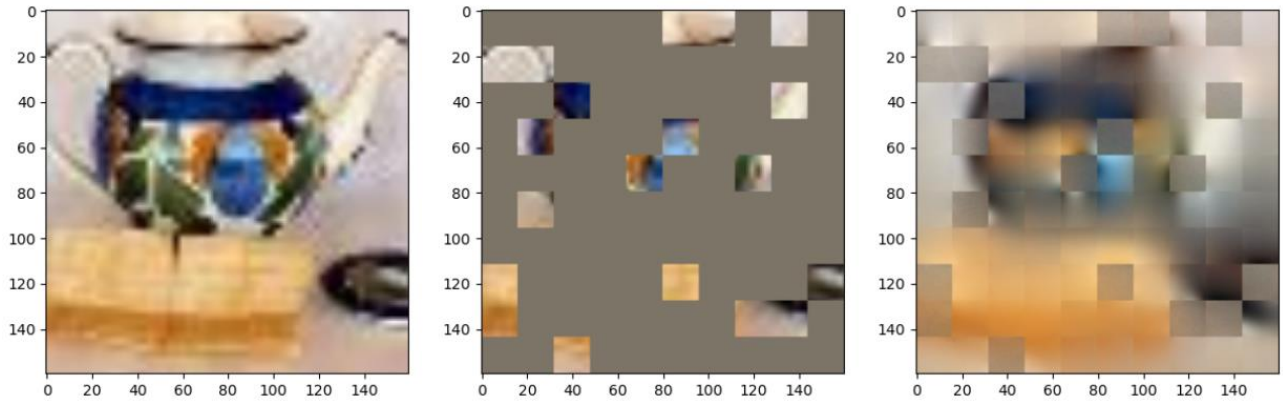


Рис. 6. Работа автоэнкодера со случайной маской

Автоэнкодер обучается методом обратного распространения ошибки, минимизируя среднеквадратичное расстояние между пикселями оригинального изображения и восстановленного декодером. Вычисление ошибки производилось только для участков, которые были скрыты маской.

Для оценки качества обучения энкодера использовался метод *linear probing*: этот подход подразумевает заморозку весов предобученной модели и обучение только классификатора. Оценка проводилась на примере решения задачи классификации на датасете *CIFAR*-10 [10]. Набор данных насчитывает 10 классов по 6000 цветных изображений размера 32×32 в каждом, обучающая выборка состоит из 50000 изображений, тестовая – из 10000 изображений.

Было проведено предобучение модели внутри масочного автоэнкодера со стратегией, использующей данные о положении объекта на изображении. Также для оценки влияния масок на

качество обучения была обучена модель внутри автоэнкодера со случайной стратегией наложения масок. В обоих случаях предобучение проводилось на 20 эпохах, последующая оценка методом *linear probing* также была на 20 эпохах.

При обучении моделированием маскированного изображения с использованием масок, опирающихся на рамки объектов была получена ошибка реконструкции 0.471. Далее предобученный энкодер при решении задачи классификации показал точность 56.46% на тестовой выборке. В свою очередь, автоэнкодер, обученный со случайной маской, достиг минимального значения ошибки реконструкции 0.384, а в задаче классификации энкодер показал точность 59.97% на тестовой выборке. Как можно видеть, качество обучения модели со случайным маскированием выше, чем с маскированием, где открытые патчи концентрируются преимущественно в районе объекта. Поскольку объекты на изображениях как правило находятся близко к центру, попытка оставлять важные части открытыми как можно чаще привела к тому, что маскируются одни и те же регионы. Полученный результат может говорить о том, что снижение разнообразия масок может оказывать на качество обучения более негативный эффект, нежели сокрытие семантически важной информации.

Заключение

В работе проведено исследование влияния стратегии наложения масок на изображения на качество обучения модели внутри масочного автоэнкодера, а также предложена стратегия, использующая метод обнаружения объектов для коррекции масок.

Выполнен обзор существующих подходов к вопросу о том, как маскировать изображения. Рассмотрены как методы, не учитывающие информацию на изображении, так и методы, которые определяют семантически важные участки и корректируют маску таким образом, они не оказались скрыты. Однако большинство таких подходов подразумевает использование дополнительной предобученной модели, что повышает ресурсоемкость обучения.

В качестве альтернативы была предложена стратегия, использующая данные, полученные алгоритмом обнаружения объектов на основе анализа численных характеристик изображений. Для удобства обучающая выборка была заранее предобработана с помощью этого алгоритма. Также были реализованы аугментация и загрузка данных, которые могли бы корректно обрабатывать сохраненную информацию о рамках.

Были реализованы и обучены масочные автоэнкодеры с двумя вариантами стратегии наложения масок: случайной и предложенной стратегии, использующей рамки объектов. Случайная стратегия показала лучшие результаты как в реконструкции изображений во время предобучения, так и в последующей оценке качества обучения энкодера на примере классификации изображений.

Список источников

1. Attention Is All You Need / Ashish Vaswani, Noam Shazeer, Niki Parmar [et al.] // Advances in Neural Information Processing Systems. – 2017. – Vol. 30.
2. Beit: Bert pre-training of image transformers / H. Bao , L. Dong, S. Piao , F. Wei // arXive.org e-Print archive. – arXiv:2106.08254 (2021).
3. Masked autoencoders are scalable vision learners / K. He, X. Chen, S. Xie [et al.] // Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. – 2022. – Pp. 16000–16009.
4. Mst: Masked self-supervised trans former for visual representation / Zhaowen Li, Zhiyang Chen, Fan Yang [et al.] // Advances in Neural Information Processing Systems. – 2021. – Vol. 34. – Pp. 13165-13176.
5. Nguyen K. B., Park C. J. Symmetric masking strategy enhances the performance of Masked Image Modeling // ICPR – 2024.
6. Selective Search for Object Recognition /J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, A. W. M. Smeulders // International Journal of Computer Vision. – 2013. – Vol.104. – Pp. 154-171.

7. Felzenszwalb P. F., Huttenlocher D. P. Efficient Graph-Based Image Segmentation // IJCV. – 2004. – Vol. 59. – Pp. 167–181.

8. Tiny-Imagenet-200 // CS231n: Deep Learning for Computer Vision. – URL: https://cs231n.stanford.edu/tiny-imagenet-200.zip (дата обращения 25.03.2025).

9. An image is worth 16x16 words: Transformers for image recognition at scale / A. Dosovitskiy, L. Beyer, A. Kolesnikov [et al.] // arXive.org e-Print archive. – arXiv:2010.11929 (2020).

10. Krizhevsky, Alex. CIFAR-10 and CIFAR-100 datasets. – URL: https://www.cs.toronto.edu/~kriz/cifar.html (дата обращения 25.03.2025).