

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
имени М. В. ЛОМОНОСОВА
МЕХАНИКО-МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ
КАФЕДРА МАТЕМАТИЧЕСКОЙ ТЕОРИИ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

Нейронные сети для сжатия изображений

**Диссертация
на соискание учёной степени**

Выполнил студент:

Сюй Цзыи
tczyi.sui@math.msu.ru

Научный руководитель:

к.ф.м.н., доцент
Часовских Анатолий Александрович

1 мая. 2022 г.

Содержание

Аннотация	2
1. Введение	2
1.1. Предыстория	2
1.2. Цель	4
1.3. Инновация	7
1.3.1. Структурные инновации	7
1.3.2. Прикладные инновации	7
1.4. Преимущество	8
1.5. Структура диссертации	8
2. Связанных с работой	10
2.1. Традиционные методы сжатия изображений	10
2.1.1. JPEG	11
2.1.2. JPEG2000	12
2.1.3. WebP	13
2.1.4. BPG	14
2.2. Глубокое обучение методы сжатия изображений	14
2.2.1. CNN	15
2.2.2. RNN	16
2.2.3. GAN	17
2.3. Другие алгоритмы обработки изображений	18
3. Нейронные сети для сжатия изображений	20
3.1. Создание областей интереса	21
3.1.1. САМ и его варианты	21
3.1.2. Состязательное дополнительное обучение	25
3.1.3. Фильтрация объектов и классов	26
3.2. Процесс сжатия	28
3.2.1. Линейное сжатие	28
3.2.2. Нелинейное сжатие	29
3.2.3. Энтропийные методы	30
3.2.4. Прогрессивное взвешивание	31
3.3. Применимость	31
3.3.1. Для лучшего отображения ROI	32
3.3.2. Для сжатия изображений	36
3.3.3. Для восстановления изображения	38
3.3.4. Применение специальных сценариев	39

4. Эксперименты	43
4.1. Набор данных	43
4.1.1. Обучающий набор	43
4.1.2. Тестовый набор	44
4.2. Критерии оценки	44
4.2.1. МГУ VQMT	45
4.2.2. EPFL VQMT	45
4.2.3. PSNR	46
4.2.4. SSIM	46
4.2.5. MS-SSIM	47
4.3. Инструменты и параметры	48
4.3.1. Инструмент повышения эффективности	48
4.3.2. Сетевая структура	49
4.4. Результаты	50
4.4.1. Мультикатегория	51
4.4.2. Две категории	52
4.4.3. Безопасность	54
4.4.4. Сжатие видео	54
5. Заключение	58
5.1. Вывод	58
5.2. Направление исследований	59
6. Список	60
Литература	61

Аннотация

Благодаря непрерывным исследованиям традиционных методов сжатия изображений и сжатия с глубоким обучением качество сжатия изображений неуклонно улучшается. Я предлагаю метод сжатия изображения, который устраниет психовизуальную избыточность и более точно сжимает изображение. Сжимайте изображение более точно, создавая семантические области интереса, то есть больше сжимайте фон и сохраняйте больше информации об объекте, тем самым улучшая общий визуальный эффект. Принят метод генерации нескольких типов отображений активации с помощью САМ и его вариантов, а качество семантических карт улучшается с помощью различных методов стирания. После получения семантической карты, в соответствии с тепловой картой, исходное изображение сжимается с различными качествами сжатия различными методами, такими как прогрессивное взвешивание. Этот метод можно комбинировать с различными фреймворками, такими как традиционные методы обучения JPEG, BPG, CNN и GAN, для получения более совершенных моделей. Я провел контролируемый эксперимент. При сжатии изображений с несколькими категориями, по сравнению с традиционными методами и некоторыми существующими методами, наши показатели субъективной оценки улучшились при той же степени сжатия. И с учетом его характеристик: сильный пояснительный, цлевой, легкий и т.д., реализовал применение трех сценариев: два типа, безопасность и сжатие видео.

глава 1

Введение

1.1. Предыстория

Сжатие изображений всегда было основной темой в области графики и обработки изображений. Глубокое обучение обладает уникальными преимуществами для извлечения объектов изображения, возможности выражения и мощности обработки многомерных данных. В настоящее время число людей, изучающих это направление, растет день ото дня, и применение глубокого обучения сжатию изображений постепенно стало одной из актуальных исследовательских проблем в настоящее время.

Благодаря постоянному совершенствованию фреймворка метод глубокого обучения для сжатия изображений постепенно созрел и был введен в эксплуатацию. Целью сжатия изображений является реализация обработки сжатия изображений путем устранения избыточности. При сжатии цифровых изображений можно определить и использовать три основных типа избыточности данных: избыточность кодирования, межпиксельная избыточность и психовизуальная избыточность. Когда один или несколько из этих трех видов избыточности уменьшаются или устраняются, достигается сжатие данных.

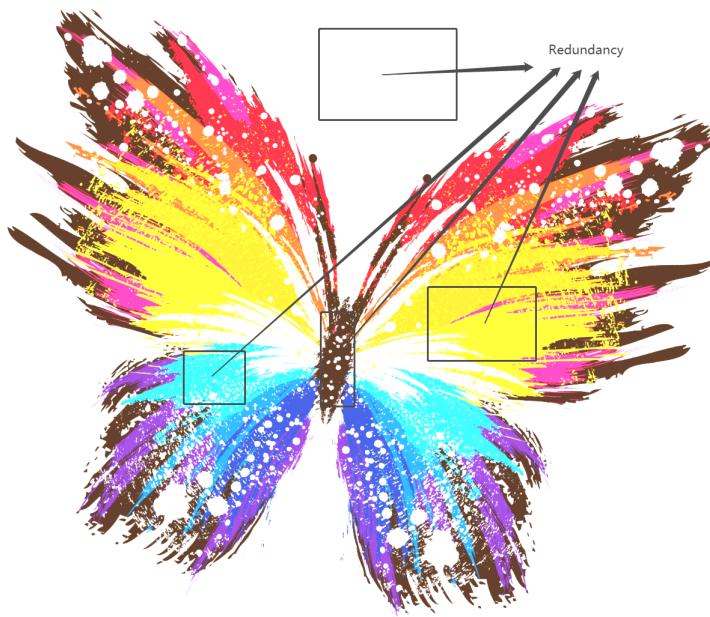


Рис. : избыточность

Широко используются традиционные стандарты кодирования изображений, такие как JPEG, JPEG2000, WebP и BPG. Традиционное сжатие изображений использует методы фиксированного преобразования и квантованные структуры

кодирования, такие как дискретные косинусоидальные преобразования и дискретные вейвлет-преобразования, которые объединяют количественную оценку и кодеры для уменьшения пространственной избыточности изображений, но не все типы изображений являются подходят для этого метода. Например, после преобразования и квантования будет эффект блока в виде блоков изображения. В то же время, из-за ограничений пропускной способности сети при передаче большого количества изображений, для достижения кодирования с низкой скоростью передачи битов изображение будет размытым. Технология глубокого обучения может оптимизировать вышеуказанные проблемы в соответствии со своими собственными характеристиками: например, с точки зрения производительности кодера технология глубокого обучения может совместно оптимизировать кодер и декодер для постоянного улучшения производительности кодера; с точки зрения четкости изображения технология сверхразрешения изображения, основанная на глубоком обучении а создание состязательных сетей может сделать реконструкцию изображений более четкой; перед лицом различных типов изображений для различных типов задач технология глубокого обучения может обеспечить более разумное и целенаправленное кодирование и декодирование изображений в соответствии с характеристиками задачи.



Рис. : эффект блока

Как традиционные, так и методы сжатия изображений с глубоким обучением основаны на разработке решения проблемы избыточности кодирования и межпиксельной избыточности, и эта статья посвящена направлению психовизуальной избыточности. Психовизуальную избыточность легче всего решить, но это также и самый сложный вид избыточности для решения.

Психовизуальная избыточность связана с реальной визуальной информацией. Она варьируется от человека к человеку. Разные люди имеют разную психовизуальную избыточность для одной и той же фотографии. Удаление избыточных психовизуальных данных неизбежно приведет к потере количественной информации, а потеря визуальной информации необратима. Точно так же, как изображение относительно мало, человеческий глаз не может непосредственно судить о его разрешении. Чтобы сжать объем данных в изображении, некоторая информация, которая не может быть непосредственно видна человеческому глазу, может быть удалена, но при увеличении изображения, которые не удаление

психовизуальной избыточности будет существенно отличаться от изображений, которые удаляют психовизуальную избыточность.

Но если мы сможем хорошо использовать этот метод, он часто может принести лучшие и неожиданные результаты.

1.2. Цель

Вы можете представить себе эти сценарии:

- Когда мы наблюдаем картину, если на ней много людей, животных или произведений искусства и архитектуры, предпочитаем ли мы наблюдать яркие объекты или окружающий фон?
- Если мы выбираем домашних животных по фотографиям в зоомагазине, фокусируемся ли мы только на этих кошках и собаках? Вместо того, чтобы обращать внимание на эти игрушки, еду и дома
- Проходя через машину или ворота, распознанные людьми, сохраненные черты лица, обращаем ли мы внимание на характеристики реальных лиц или обнаруживаем эти фальшивые лица?
- Для видеозаписей, записанных камерами на транспортных узлах, мы обращаем внимание только на траекторию движения транспортных средств, игнорируя номерные знаки, цвета, погоду и другие дорожные условия.

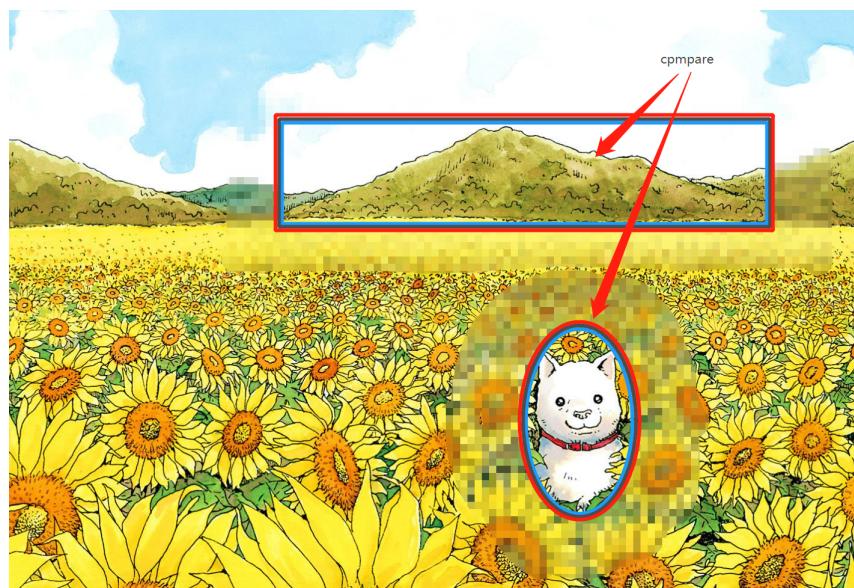


Рис. : сравнение

Подумайте над следующими вопросами:

- Наблюдая за изображением, которое было сжато, где находится основной фокус просматриваемой информации? Или какая доля области, на которую вы смотрите большую часть времени, составляет всю картину?

- Возьмите цель, которую необходимо соблюдать, чтобы повторно просмотреть эту картинку. Отличается ли визуальный эффект от ощущения просмотра в начале? Или мы сжимаем область изображения в разной степени. Если разные люди будут наблюдать, получат ли они разные выводы о степени сжатия изображения?

Это показывает два момента: информация, наблюдаемая первым человеческим глазом, представляет собой лишь небольшую часть изображения, а область, наблюдаемая вторым, сильно связана с целью. Исходя из этого, прогрессивное сжатие выполняется путем построения области интереса, то есть методом сжатия, который сжимает больше фона и меньше объектов.

Поэтому при выполнении сжатия изображения его можно разделить на две подзадачи: одна заключается в построении интересующей области, а другая - в выполнении постепенного сжатия в соответствии с отображением и заранее установленными правилами.

С точки зрения цели, мы можем построить карту интересующей области с помощью обнаружения объектов или семантической сегментации. Обнаружение объектов позволяет хорошо распознавать области объектов, а семантическая сегментация позволяет хорошо разделять изображения на различные объекты и фоны.

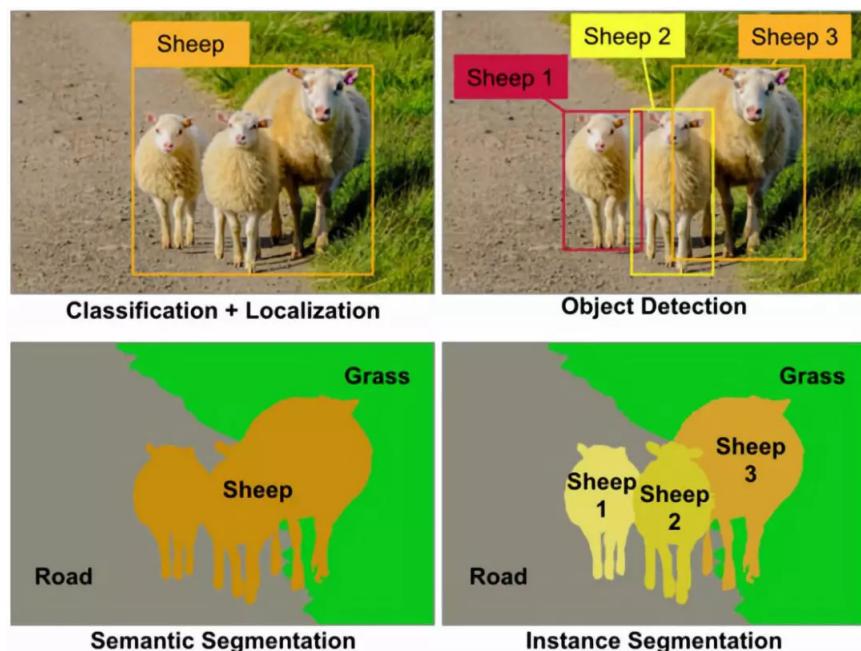


Рис. : обработка изображений

Но это будет иметь следующие проблемы:

1. Результатом обнаружения цели является область, а результатом семантической сегментации является разделительная линия. Нам нужно влияние градиентов, а не простое усечение; во-первых, существуют разные области с разными степенями важности внутри объекта, а сам объект имеет важное

и неважное значение; во-вторых, пространство вокруг объекта будет более важным, чем весь фон. Если они рассматриваются как фон, некоторая важная информация может быть отфильтрована, что приведет к пустой трате информации.

2. Реализация обнаружения объектов и семантической сегментации очень сложна. Мы не согласны превращать простую задачу сжатия изображений в более сложную задачу для решения; обнаружение объектов и семантическая сегментация потребляют слишком много вычислительных ресурсов, что, очевидно, повлияет на степень популярности; и это очень зависит от точность и обобщение модели или алгоритма. Если выходной результат нестабилен, это приведет к особенно большому количеству цепных реакций, что сделает невозможным точное объяснение полученных результатов сжатия.

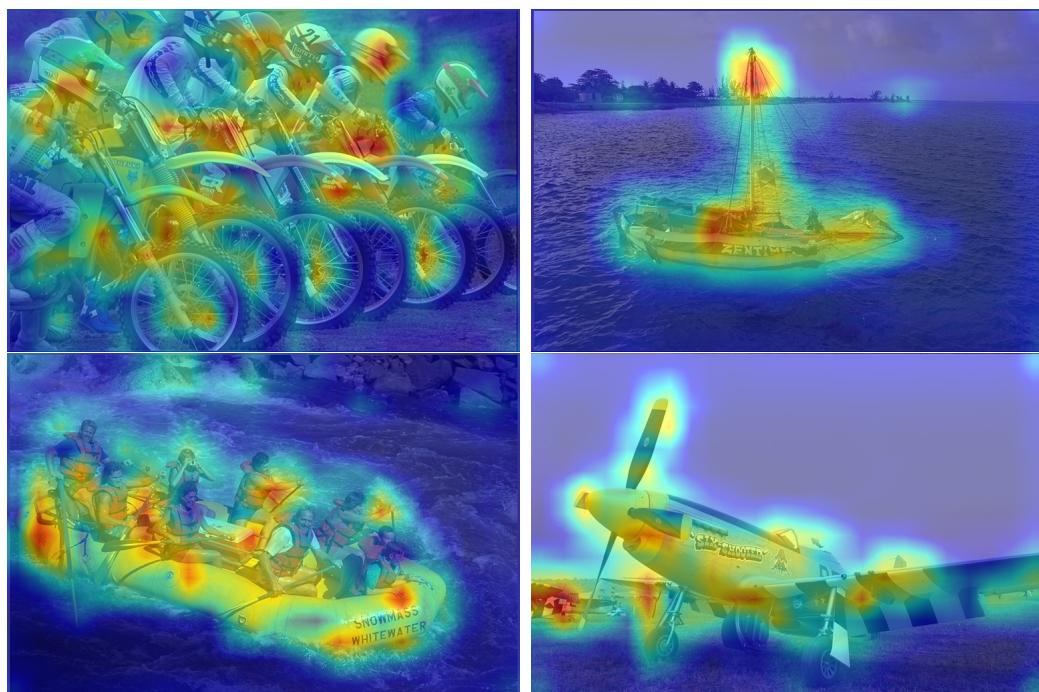


Рис. : карта активации класса

Итак, существует ли простая классификационная сеть, которая может соответствовать этим требованиям? В то же время это также может гарантировать, что сжатие будет постепенным, а не дискретным, что очень важно. Поэтому я предложил диаграмму активации классов на основе САМ и ее вариантов для сжатия изображений. ROI, полученный с помощью фреймворка, может обнаруживать объекты и сегментировать семантику, и в то же время он может давать относительно непрерывное разделение изображений. И для спроектированной сети мы реализовали применение нескольких направлений или сценариев, упомянутых выше.

1.3. Инновация

Этот метод принципиально отличается от традиционных методов сжатия и глубокого обучения.

1.3.1. Структурные инновации

Для обучения многокатегориальной нейронной сети для генерации ROI используются различные фреймворки сам и его вариантов. Например, GradCAM используется для реализации сам-реализации сети классификации с вычислением градиента, а также используется разделяемая по глубине свертка MSROI. На этой основе генерируются карты активации нескольких категорий. с помощью линейных, нелинейных, многокатегорийных методов взвешивания и энтропии, а также улучшенных многокатегорийных карт активации генерируются путем дополнительного изучения признаков, а затем реализуется оптимизация сжатия психовизуальных изображений с использованием прогрессивного сжатия и других методов.

1.3.2. Прикладные инновации

В отличие от простого улучшения качества сжатия, психовизуальное сжатие изображений может использоваться в нескольких специальных сценариях, что делает сжатие изображений не только приложением в направлении обработки изображений, но и может удовлетворить потребности повседневной жизни. Благодаря непрерывным экспериментам мы достигли следующих применений: оптимизация сжатия, адаптированная к общим изображениям, сжатие только для специальных целей, сжатие конфронтации безопасности изображения, сжатие видео для специальных сцен и методы сжатия в сочетании с CNN и GAN.



Рис. : две категории

1.4. Преимущество

- Улучшение качества субъективного сжатия: В соответствии с семантическим сжатием для фона и объектов выполняются различные степени сжатия. При условии стабильного объективного качества субъективное качество сжатия может быть значительно улучшено.
- Интерпретируемость: В отличие от сквозных нейронных сетей, используемых для сжатия изображений, получение карт объектов для сжатия поддается интерпретации и больше не является операцией черного ящика.
- Высокая маневренность и малый вес: Он может классифицировать нейронные сети для CNN и обнаруживать все объекты с характерными структурами, вместо того, чтобы получать ROI только для одного объекта, и он может адаптироваться к различным требованиям к степени сжатия с помощью настройки параметров сети и меток, гибкий, высокая доступность и малый вес.
- Целенаправленное обучение: можно получить обучение для конкретных наборов данных, конкретных приложений сжатия.
- Сильная практичность: он основан на формальных инновациях без изменения исходной структуры сжатия, поэтому, в принципе, любая модель сжатия, такая как JPEG, BPG, CNN, GAN и т.д., Может быть объединена.
- Прогрессивность: отличается от обнаружения цели и семантической сегментации, с четкими границами сегментации, прогрессивной и непрерывной с лучшим субъективным качеством сжатия, потерей сжимаемого состояния области, не относящейся к объекту, в области обнаружения цели или важностью граничной области вокруг объекта в семантической сегментации.
- Специальные цели, такие как использование семантики в GAN, одна из которых заключается в том, что разрешение восстановленного изображения может быть улучшено с помощью семантики, а другая заключается в том, что дискриминатор D в модели может быть добавлен во время обучения, что может помочь в обучении генератора G.

1.5. Структура диссертации

Эта статья разделена на пять основных глав. Первая глава представляет собой введение. В ней в основном описываются предпосылки исследования, причины, цели и значение этой статьи, а также кратко описываются модели, методы и инновации. Во второй главе будут рассмотрены традиционные методы

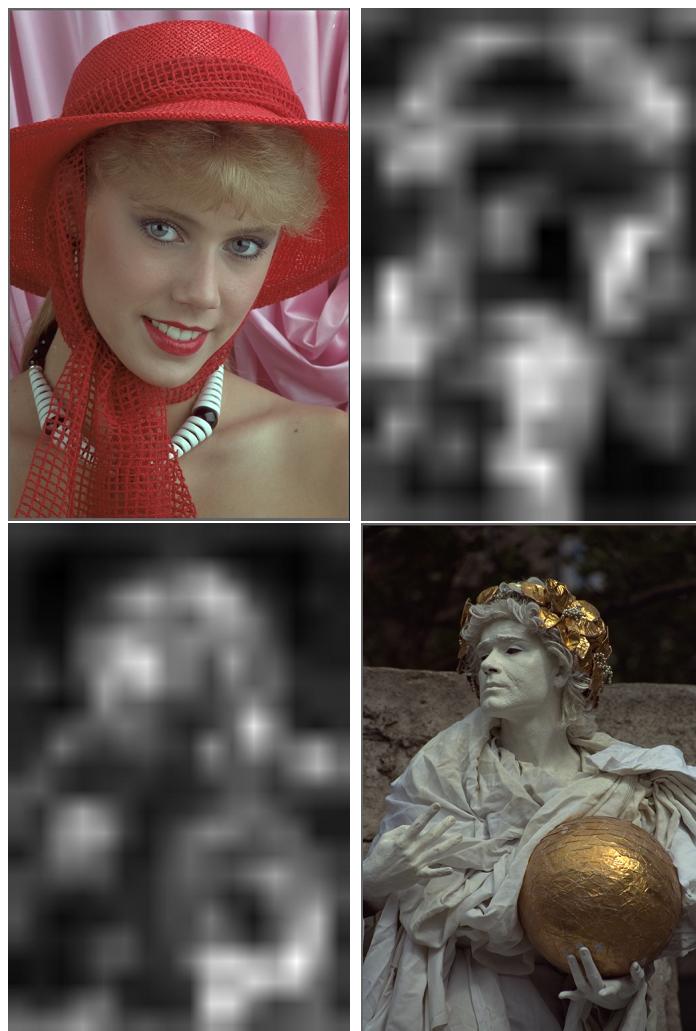


Рис. : область интереса

и базовые знания о глубоком обучении, которые необходимо применять. В третьей главе в основном представлены основные принципы модели и формулы в методах исследования, а также указаны применимые сценарии, преимущества и недостатки каждого метода. В третьей главе в основном описывается выбор обучающих наборов и наборов для проверки, а затем выбирает метод оценки, суммирует параметры обучения и параметры программы. Наконец, перечислены экспериментальные данные и эффект модели анализа. В приложении эффект в различных ситуациях. Четвертая глава представляет собой основной вывод и меры, которые могут быть улучшены для исправления методов исследования в этой статье, а также направления исследований, предусмотренные в будущем.

глава 2

Связанных с работой

Разработанная мной модель представляет собой комбинацию глубокого обучения и традиционных методов. Метод обучения классификации CNN используется для генерации функции отображения, необходимой для сжатия. И используется BPG, традиционный метод сжатия с наилучшим эффектом. Моя модель основана на методологических инновациях, но по сути это уже существующий популярный метод сжатия. На основе методологических исследований было усовершенствовано, добавлено больше контента для реализации применения определенных требований. Поэтому еще необходимо понимать содержание некоторых традиционных методов и методов глубокого обучения.

Технология глубокого обучения может оптимизировать вышеуказанные проблемы в соответствии со своими собственными характеристиками: например, с точки зрения производительности кодера технология глубокого обучения может совместно оптимизировать кодер и декодер для постоянного улучшения производительности кодера; с точки зрения четкости изображения технология сверхразрешения изображения, основанная на глубоком обучении а создание состязательных сетей может сделать реконструкцию изображений более четкой; перед лицом различных типов изображений для различных типов задач технология глубокого обучения может обеспечить более разумное и целенаправленное кодирование и декодирование изображений в соответствии с характеристиками задачи.

2.1. Традиционные методы сжатия изображений

Широко используются традиционные стандарты кодирования изображений, такие как JPEG, JPEG2000, WebP и BPG. Традиционное сжатие изображений использует методы фиксированного преобразования и квантованные структуры кодирования, такие как дискретные косинусоидальные преобразования и дискретные вейвлет-преобразования, которые объединяют количественную оценку и кодеры для уменьшения пространственной избыточности изображений. И сжатие изображений, основанное на глубоком обучении, не является независимым от традиционных методов сжатия изображений.

2.1.1. JPEG

Метод сжатия JPEG¹ обычно представляет собой сжатие с потерями. Алгоритм кодирования сжатия JPEG базовой системы разделен на 11 шагов: преобразование цветового режима, выборка, разбиение на блоки, дискретное косинусное преобразование DCT, сортировка по зигзагообразному сканированию, количественное определение, кодирование коэффициентов постоянного тока с дифференциальной импульсной модуляцией, вычисление коэффициентов постоянного тока в промежуточном формате, кодирование коэффициентов переменного тока, энтропийное кодирование. Среди них, при использовании JPEG, основными шагами, на которые все обращают наибольшее внимание, являются: DCT, квантование и энтропийное кодирование.

Если данные изображения находятся в формате RGB перед сжатием, формат необходимо преобразовать в формат YUV. Y представляет компонент яркости, а UV представляет компонент цветности. Далее данные делятся на макроблоки размером 8x8 или 16x16.

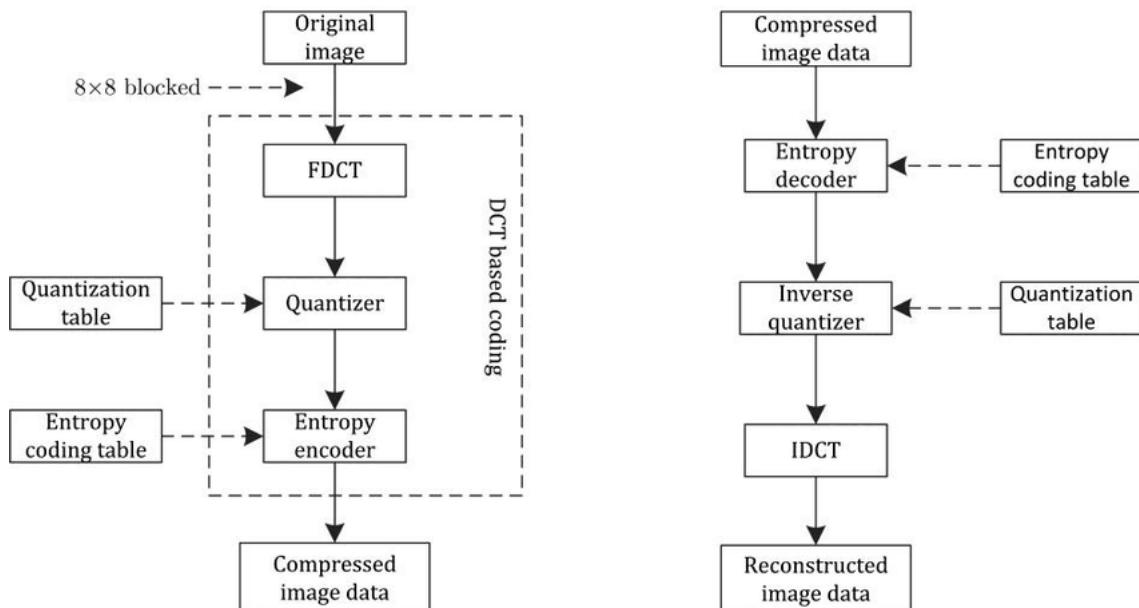


Рис. : JPEG flow

1. Дискретное косинусное преобразование: Проще говоря, дискретное косинусное преобразование - это матричная операция. После дискретного косинусного преобразования высокочастотные данные и низкочастотные данные разделяются. В левом верхнем углу матрицы находятся высокочастотные данные, которые имеют большие значения, а в правом нижнем углу - низкочастотные данные, которые имеют меньшие значения.

¹Wallace G K, 1991. The JPEG still picture compression standard.

2. Квантовая операция: Квантования является основным этапом потери данных при сжатии. Ее основной принцип заключается в делении каждого значения в макроблоке после преобразования DCT на соответствующий коэффициент в таблице квантования и округлении его. Среди них коэффициенты, соответствующие высокочастотной части квантованной таблицы, намного больше, чем коэффициенты низкочастотной части. После квантования частотные коэффициенты высокочастотной части значительно ослабляются или даже многие очищаются до нуля, в то время как частотные коэффициенты низкочастотной части значительно уменьшаются. Поскольку человеческий глаз более чувствителен к низкочастотной части, она квантуется, а затем восстанавливается в изображение, что оказывает меньшее влияние на визуальный эффект, но данные эффективно сжимаются. Конечной целью количественной оценки является уменьшение амплитуды ненулевых коэффициентов в низкочастотной части и увеличение числа коэффициентов с нулевым значением в высокочастотной части.
3. Энтропийное кодирование: Это включает в себя расположение компонентов изображения шрифтом Z. Чтобы облегчить последующее кодирование, блоки данных необходимо переставить перед кодированием, чтобы данные в низкочастотной части были ранжированы первыми, а данные в высокочастотной части были ранжированы вторыми, чтобы увеличить количество последовательных нулевых значений в массиве, так что используется Z-образное расположение. а затем использование кодировки Хаффмана для всего остального.
4. Декодирование: Процесс декодирования отличается от процесса кодирования, и изображение восстанавливается с помощью ряда операций, таких как обратное энтропийное кодирование, обратное квантование и обратное дискретное косинусное преобразование.

2.1.2. JPEG2000

Самое большое различие между JPEG2000² и JPEG заключается в использовании дискретного вейвлет-преобразования. Кроме того, были добавлены некоторые этапы предварительной обработки.

Дискретное вейвлет-преобразование дискретизирует масштаб и преобразование базовых вейвлетов. При обработке изображений двоичные вейвлеты часто используются в качестве функций вейвлет-преобразования, то есть они делятся на целые степени 2.

Косинус-преобразование-это классический инструмент спектрального анализа. Он рассматривает частотной характеристики весь процесс временной области или временной области характеристик весь процесс частотной области.

²Rabbani M, Joshi R, 2002. An overview of the JPEG 2000 still image compression standard.

Таким образом, для плавного процессов, он имеет неплохие результаты, но для негладких процессов, он обладает множеством недостатков. В крайних случаях, изображений JPEG сохраняют только основная информация, которая отражает внешний вид изображения, а мелкие детали изображения будут потеряны. Вейвлет-преобразование является современным инструментом спектрального анализа. Его можно исследовать как в частотной характеристики локальных процессов домен и домен характеристики местного частотной области процессов. Поэтому, даже для нестационарных процессов, легко обрабатывать. Он может превратить изображение в серии вейвлет-коэффициентов, которые могут быть эффективно сжимаются и сохраняются. Кроме того, острые углы вейвлетов может лучше представить образ, потому что он устраняет ДКП. Сжатие, как правило, имеет квадратную эффект.

2.1.3. WebP

WebP³ с потерями сжимает данные изображения на основе метода прогнозирующего кодирования в кодировании видео VP8. Основные этапы аналогичны сжатию JPEG, которое в основном включает преобразование формата, сегментацию подблоков, прогнозирующее кодирование, FDCT, количественное определение, Z-расположение и энтропийное кодирование.

FDCT - Forward Discrete Cosine Transform, заключается в преобразовании набора пикселей в пространственной области в коэффициенты в частотной области и выполнении FDCT для каждого макроблока, так что низкочастотная часть преобразованных данных распределяется в верхнем левом углу блока данных, и высокочастотная часть сосредоточена в правом нижнем углу. Первый коэффициент в верхнем левом углу называется коэффициентом постоянного тока, а остальные - коэффициентами переменного тока.

В WebP в качестве метода энтропийного кодирования используется логическое арифметическое кодирование. Отличие от других методов энтропийного кодирования заключается в том, что другие методы энтропийного кодирования обычно делят входное сообщение на символы, а затем кодируют каждый символ, в то время как арифметическое кодирование непосредственно кодирует все входное сообщение в число, десятичное число n , которое удовлетворяет. Чем длиннее сообщение, тем меньше интервал между его кодированием и указанием, и тем больше двоичных битов требуется для указания этого интервала.

Почему WebP с потерями лучше, чем JPEG? Основная причина - прогностическое кодирование. Адаптивное разделение на части также обеспечивает более высокую производительность. Циклическая фильтрация очень помогает в случае средних и низких скоростей передачи данных. По сравнению с арифметическим кодированием Хаффман увеличил производительность сжатия на 5-10%.

³Web. WebP Image format. <https://developers.google.com/speed/webp/>

2.1.4. BPG

BPG⁴, полное название Better Portable Graphics - Лучшая портативная графика, это формат, который утверждает, что он более популярен, чем текущий JPEG. Лучший формат сжатия сжатие изображения. Формат кодирования изображений основан на Высокоэффективном кодировании видео HEVC внутрикадрового кодирования Усовершенствованная технология. BPG является подмножеством HEVC, но информация заголовка, используемая HEVC для видео, заменена более упрощенной информацией заголовка изображения BPG, что означает, что BPG по существу использует технологию HEVC и соответствует спецификациям HEVC.

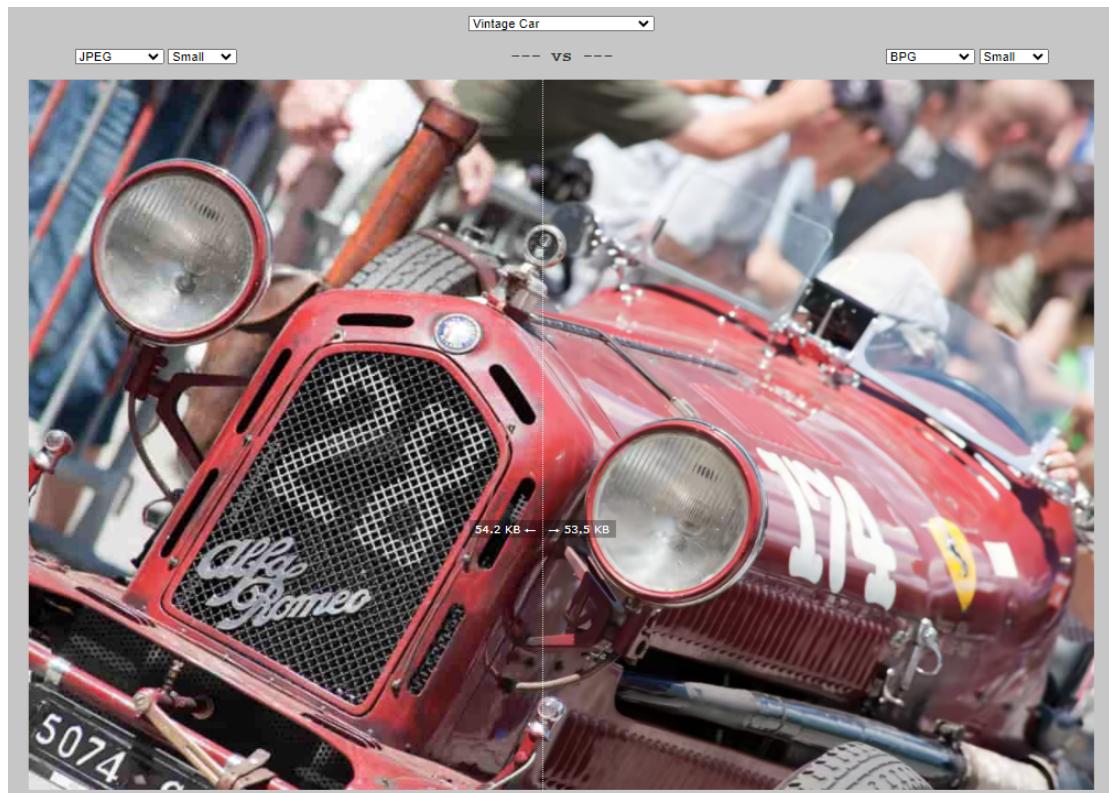


Рис. : BPG vs JPEG

С точки зрения эффекта сжатия BPG лучше чем WebP, но патентные сбои, связанные с ядром HEVC, используемым BPG, не позволили ему широко использоваться на рынке.

2.2. Глубокое обучение методы сжатия изображений

Традиционное сжатие изображений использует методы преобразования для сжатия изображений в сочетании с энтропийным кодированием, но не для всех типов изображений подходят для этого метода. Например, после преобразования и квантования будет эффект блока в виде блоков изображения. В то же

⁴Fabrice Bellard. BPG Image format. <https://bellard.org/bpg/>

время, из-за ограничений пропускной способности сети при передаче большого количества изображений, для достижения кодирования с низкой скоростью передачи битов изображение будет размытым.

Технология глубокого обучения может оптимизировать вышеуказанные проблемы в соответствии со своими собственными характеристиками: например, с точки зрения производительности кодера технология глубокого обучения может совместно оптимизировать кодер и декодер для постоянного улучшения производительности кодера; с точки зрения четкости изображения технология сверхразрешения изображения, основанная на глубоком обучении а создание состязательных сетей может сделать реконструкцию изображений более четкой; перед лицом различных типов изображений для различных типов задач технология глубокого обучения может обеспечить более разумное и целенаправленное кодирование и декодирование изображений в соответствии с характеристиками задачи.

Глубокое обучение использует сквозное структурное проектирование и различные типы сетевого обучения для замены традиционных фиксированных методов преобразования, тем самым улучшая сжатие изображений. В то же время быстрое развитие графических процессоров в последние годы обеспечило вычислительные гарантии для проектирования более разнообразных сетевых структур, а также обеспечило аппаратную поддержку для повышения производительности, что позволило улучшить сжатие изображений на основе глубокого обучения во всех аспектах, таких как разрешение и скорость передачи данных.

Методы сжатия изображений, основанные на глубоком обучении, в основном представляют собой сжатие изображений с потерями, основанное на мощных возможностях моделирования глубокого обучения. Производительность сжатия изображений, основанная на глубоком обучении, превзошла производительность JPEG и BPG, и этот разрыв в производительности все еще постепенно увеличивается. Далее представим сверточную нейронную сеть CNN, рекуррентную нейронную сеть GRNN и генеративно-состязательную сеть GAN.

2.2.1. CNN

Характеристики самого CNN⁵ также могут напрямую заменять кодеры и декодеры при сжатии изображений. Две характеристики разреженного соединения и совместного использования параметров в операциях свертки CNN позволяют CNN демонстрировать преимущества при сжатии изображений. Разреженные связи могут ограничивать количество выходных параметров размером ядра свертки. В изображении существует пространственная структура организации. Пиксель в изображении имеет тесную пространственную связь с окружающими пикселями. Разреженные соединения используют эту связь и принима-

⁵Liu H, Chen T, Shen Q, 2018. Deep Image Compression via End-to-End Learning. <https://arxiv.org/abs/1806.01496>

ют только взаимосвязанные области в качестве входных пикселей. После таким образом, локальная информация, получаемая всеми нейронами, интегрируется в более глубокую сеть, и может быть получена глобальная информация, тем самым уменьшая параметры и уменьшая сложность вычисления. Распределение веса означает, что параметры каждого нейрона одинаковы, и параметры совместно используются при обработке изображений одного и того же сверточного ядра. Используя этот метод, сверточные нейронные сети значительно сократят количество параметров и в определенной степени избегут переобучения. Эти две характеристики сверточных нейронных сетей лучше снижают сложность вычислений, так что обучение может развиться до более глубокой и совершенной структуры сети. В то же время эти две характеристики также уменьшают объем данных, сжатых изображением. Сжатие изображений CNN в основном выполняет сквозное сжатие изображений. С помощью CNN разрабатываются терминалы кодирования и декодирования, и благодаря большому объему данных изображения и оптимизированным сетевым методам достигается высокопроизводительная структура сжатия.

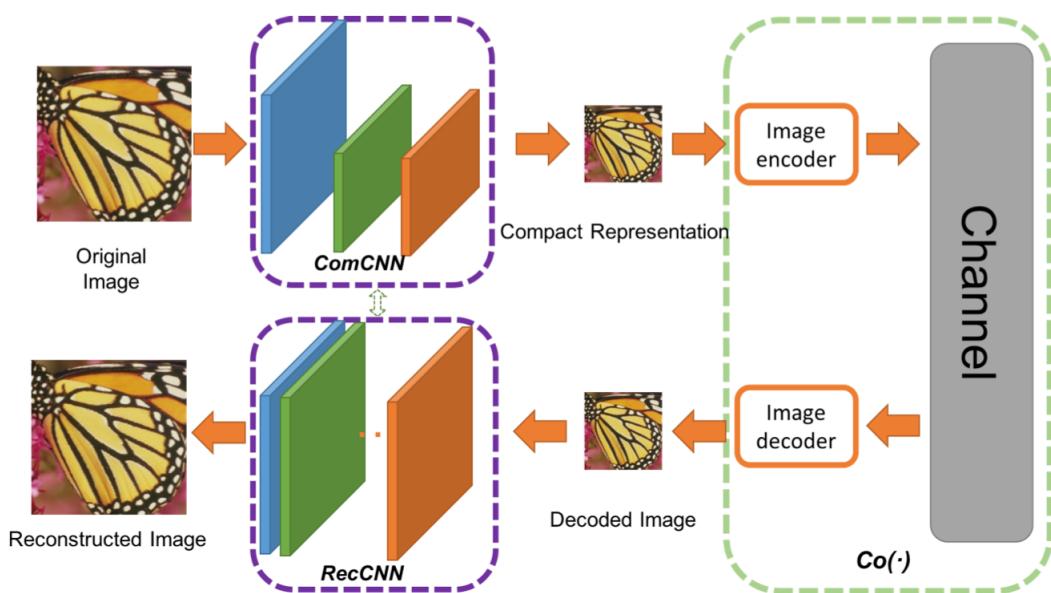


Рис. : CNN

2.2.2. RNN

Сначала RNN⁶ не получил широкого распространения из-за трудностей с реализацией. После этого, с улучшением структуры RNN и улучшением производительности графического процессора, RNN постепенно стал популярным. По сравнению с CNN, RNN и CNN имеют одинаковые характеристики совместного

⁶Toderici G, Vincent D, Johnston N, 2017. Full Resolution Image Compression with Recurrent Neural Networks. <https://arxiv.org/abs/1608.05148>

использования параметров. Разница в том, что совместное использование параметров CNN является пространственным, в то время как RNN основано на времени, то есть на последовательности. Это делает RNN “памятью” предыдущей информации о последовательности, а его метод обучения итеративно вычисляется методом градиентного спуска. Один из этих двух методов может улучшить степень сжатия данных, а другой заключается в том, что скоростью передачи битов изображения можно управлять итеративно, что может улучшить производительность сжатия изображения. Таким образом, сжатие изображений с использованием RNN позволило достичь относительно хороших результатов в сжатии изображений с полным разрешением и управлении скоростью передачи данных степени сжатия, но стоит отметить, что при использовании RNN большинству из них необходимо ввести LSTM или GRU для решения проблемы долгосрочной зависимости, поэтому обучение модели будет сложнее.

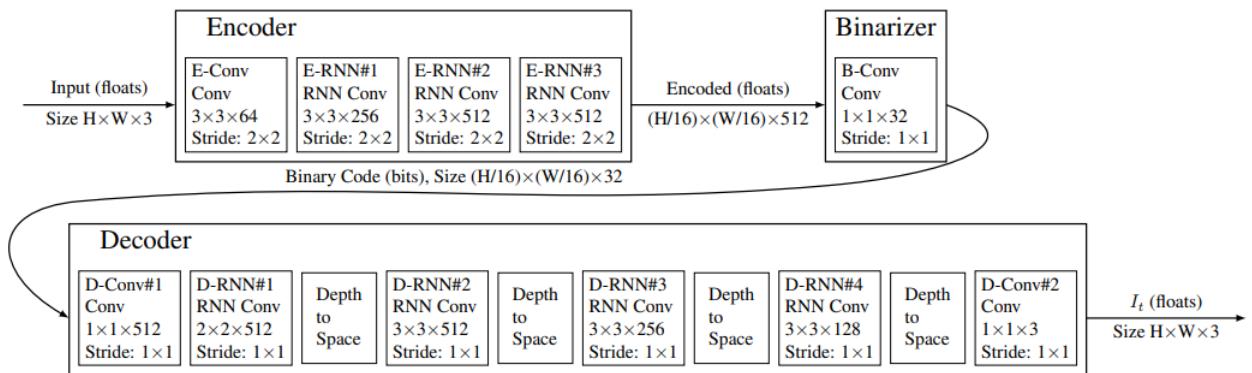


Рис. : RNN

2.2.3. GAN

Нелегко получить изображения высокой четкости и корректно реконструированные с помощью GAN⁷. Обучение GAN сложнее. Во время обучения необходимо координировать уровень обучения генератора и дискриминатора. Если дискриминатор обучен слишком хорошо, это приведет к возникновению у генератора таких проблем, как исчезновение градиента во время обучения и когда степень обучения дискриминатора недостаточна, это приведет к тому, что генератор не сможет сгенерировать идеальное изображение. Чтобы получить сгенерированные изображения с более высоким разрешением, предложили алгоритм для генерации восстановленных изображений с высоким разрешением из сопоставления семантических меток. Алгоритм не только обеспечивает экстремальное сжатие при сверхнизких скоростях передачи битов при условии

⁷Tschannen M, Agustsson E, Lucic M, 2018. Deep Generative Models for Distribution-Preserving Lossy Compression. <https://arxiv.org/abs/1805.11057>

полного разрешения, но также реализует восстановленные изображения с высоким разрешением при низких скоростях передачи.

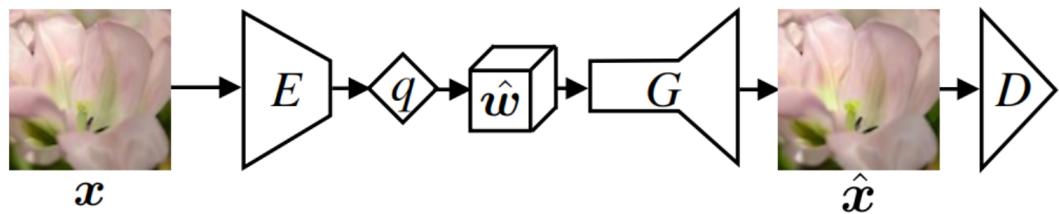


Рис. : GAN

Сжатие изображений на основе GAN также имеет много преимуществ: во-первых, GAN может сжимать изображения с полным разрешением, что показывает, что этот метод имеет хорошую применимость; во-вторых, GAN может достигать сжатия изображений при экстремальных скоростях передачи битов; и в-третьих, хотя изображения, созданные GAN могут возникнуть проблемы, преимущества разрешения и четкости восстановленных изображений заслуживают внимания.

2.3. Другие алгоритмы обработки изображений

CNN реализует три поэтапных уровня изображений:

Классификация

Классификация заключается в структурировании изображения в определенную категорию информации и описании изображения с помощью заранее определенной категории или идентификатора экземпляра.

Обнаружение

Задача классификации заботится о целом и дает описание содержимого всего изображения, в то время как обнаружение фокусируется на конкретной цели объекта и требует одновременного получения информации о категории и информации о местоположении этой цели.

Сегментация

Сегментация включает в себя семантическую сегментацию и сегментацию экземпляра. Первая представляет собой расширение разделения переднего фона, что требует разделения частей изображения с разной семантикой, в то время как вторая представляет собой расширение задачи обнаружения, которая требует описания контура цели (который более тонкий, чем рамка обнаружения).

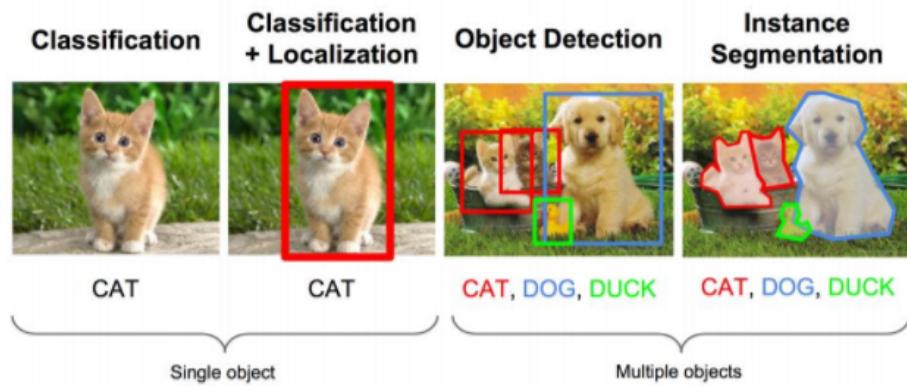


Рис. : обработка изображений

Эти обработки на основе изображений может быть использована для сжатия изображений. Фактически, в этой статье также используется метод извлечения объектов изображения для сжатия изображения. Среди них извлечение признаков достигается путем обучения классификационной сети. В то же время извлеченные объекты также могут быть использованы для обнаружения цели и семантического сжатия. В следующей главе я подробно представлю свою модель и объясню, как сочетать ее с методами, описанными в этой главе.

глава 3

Нейронные сети для сжатия изображений

Фреймворк в основном разделен на две части: создание областей интереса и процесс сжатия. Как показано на рисунке, верхняя часть представляет собой извлечение диаграммы активации класса классификационной сети, а нижняя часть представляет собой сжатие изображения в соответствии с ROI. В этой главе состав каждого произведения подробно объясняется отдельно.

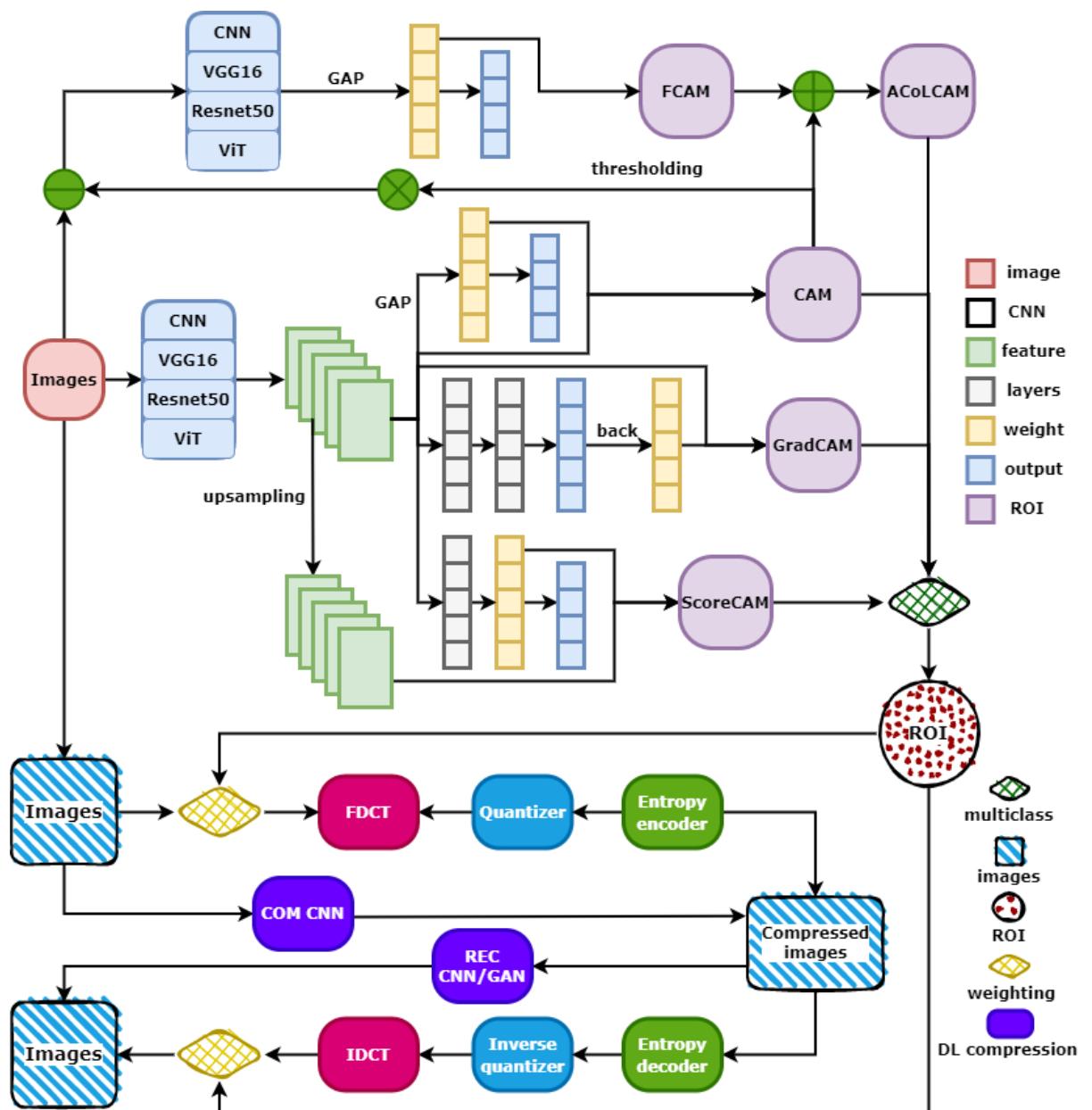


Рис. : технологическая схема

3.1. Создание областей интереса

3.1.1. CAM и его варианты

CAM

Первым, кто предложил идею GAP¹, была статья под названием "Сеть в сети". В этой статье было обнаружено, что использование GAP вместо полностью подключенного слоя может не только уменьшить размер, предотвратить переобучение и уменьшить большое количество параметров. Производительность сети также очень хорошая.

Если мы используем GAP вместо FC, преимущество заключается в минимизации количества параметров при сохранении высокой производительности, структура становится простой, и можно избежать переоснащения. Однако недостатком является то, что по сравнению с FC скорость конвергенции зазора ниже.

В статье "Изучение глубоких функций для дискриминационной локализации" они обнаружили роль GAP, который может сохранять пространственную информацию и определять ее местоположение.

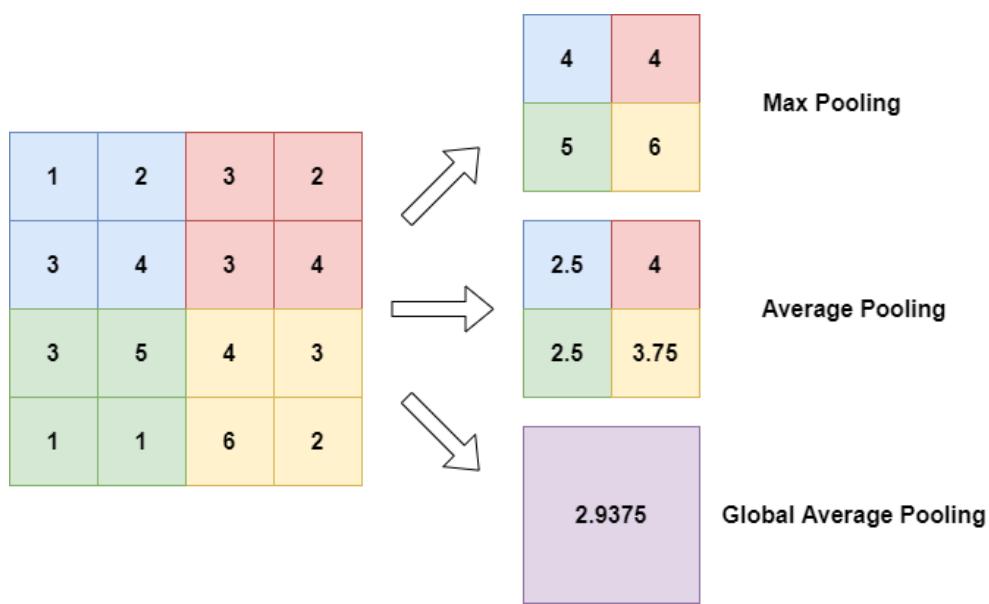


Рис. : Global Average Pooling

Итак, что такое сопоставление активации класса? CAM - это инструмент, который помогает нам визуализировать CNN. Используя CAM, мы можем четко наблюдать, на какой области изображения фокусируется сеть. Например, наша сеть распознает эти две картинки, на одной из которых изображена кошка, а на другой - собака. С помощью инструмента CAM мы можем четко видеть, на какой части изображения фокусируется сеть и на какой части результата получается результат.

¹Zhou B, Khosla A, Lapedriza A, 2015. Learning Deep Features for Discriminative Localization. <https://arxiv.org/abs/1512.04150>

Использую слова в статье, диаграммы активации классов представляют собой всего лишь взвешенные линейные суммы этих визуальных шаблонов, которые существуют в разных пространственных местоположениях. Просто увеличив дискретизацию карты активации класса до размера входного изображения, мы можем определить область изображения, которая наиболее соответствует определенной категории. Если перевести этот отрывок на математический язык, то получится следующая формула.

$$Y_{CAM}^c = \frac{1}{Z} \sum_k w_k^c \sum_i \sum_j A_{ij}^k$$

Для создания САМ требуется сеть CNN, основанная на обучении классификации. Вначале это ввод, а в середине много сверточных слоев. После последнего сверточного слоя следует глобальный средний объединяющий слой и, наконец, слой softmax для получения выходных данных. GAP предназначен для преобразования карты объектов в вектор объектов, и каждый слой карты объектов представлен значением. Мы умножаем вес, соответствующий требуемому классу, на слой, соответствующий карте объектов, и нормализуем его с помощью тепловой карты, то есть следующей строки тепловых карт F_n :

$$W_1 * F^1 + W_2 * F^2 + \dots + W_n * F^n = CAM$$

Таким образом, САМ представляет собой взвешенную линейную сумму. Вообще говоря, размер последнего слоя сверточного слоя не будет равен входному размеру, поэтому нам нужно увеличить выборку этого типа карты активации до размера исходного изображения, а затем наложить его на исходное изображение, вы можете наблюдать, на какой области изображения фокусируется сеть когда он получит этот результат. Это означает, что размер входного изображения и глубина слоя свертки могут быть введены произвольно.

Depthwise

САМ очень полезен, но есть некоторые недостатки. Прежде всего, мы должны изменить структуру сети, например, заменить полностью подключенный уровень на глобальный средний уровень объединения, что не способствует обучению. Во-вторых, это метод визуализации, основанный на задачах классификации, и он может не иметь такого хорошего эффекта при использовании для задач регрессии.

Глубокая разделяемая свертка используется в MSROI для решения этой проблемы вместо GAP. Ядро свертки Глубинной свертки отвечает за канал, а канал свертывается только одним ядром свертки.

Для трехканального цветного входного изображения размером 5*5 пикселей свертка по глубине сначала проходит первую операцию свертки, и DW полно-

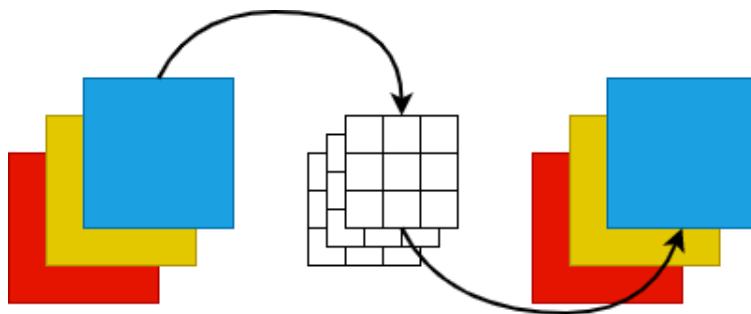


Рис. : Depthwise Convolution

стью выполняется в двумерной плоскости. Количество сверточных ядер совпадает с количеством каналов в предыдущем слое. Таким образом, трехканальное изображение вычисляется для создания 3 карт объектов, как показано на рисунке ниже.

Количество карт объектов после завершения свертки по глубине совпадает с количеством каналов во входном слое, и карта объектов не может быть расширена. Более того, эта операция выполняет независимые операции свертки на каждом канале входного уровня и неэффективно использует информацию о характеристиках разных каналов в одном и том же пространственном положении. Поэтому для объединения этих карт объектов для создания новой карты объектов часто требуется точечная свертка.

В некоторых облегченных сетях, таких как mobilenet, будет использоваться глубинная разделяемая свертка depthwise separable convolution, состоящая из двух частей: depthwise и pointwise, которые объединяются для извлечения карт объектов. Так что это все равно не самый лучший способ.

GradCAM

Также для решения первой проблемы появилась усовершенствованная технология под названием Grad-CAM². Grad-CAM можно визуализировать без изменения структуры сети. Grad-CAM решает эту проблему. Основная идея та же, что и у CAM. Он также получает веса, соответствующие каждой паре карт объектов, и, наконец, находит взвешенную сумму. Разница заключается в процессе вычисления весов. CAM получает веса путем замены полностью связанным слоем РАЗРЫВА и переподготовки, в то время как Grad-CAM находит другой способ вычисления весов, используя глобальное среднее значение градиентов. Фактически, после строгого математического вывода веса, рассчитанные с помощью Grad-CAM и CAM, эквивалентны.

Сначала определите вес важности сопоставления объектов к классификацией с как:

²Selvaraju R, Cogswell M, Das A, 2016. Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. <https://arxiv.org/abs/1610.02391>

$$\alpha_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\partial y^c}{\partial A_{ij}^k}$$

Вес взвешивается на последнем слое карты объектов, объединяется линейно и подается в функцию активации ReLU для получения:

$$L_{\text{Grad-CAM}}^c = \text{ReLU} \left(\sum_k \alpha_k^c A^k \right)$$

Причина, по которой используется ReLU, заключается в том, что мы заботимся только о позициях, которые оказывают положительное влияние на классификацию, а те позиции, которые получают отрицательные числа, с большей вероятностью принадлежат к другим категориям, которые не относятся к категории С.

Однако Grad-CAM не может хорошо определить местоположение нескольких целей одного и того же типа. Результатом невзвешенной частной производной является то, что он может определить местоположение только части объекта.

ScoreCAM

Score-CAM³ отличается от предыдущего метода, основанного на отображении активации класса. Score-CAM избавляется от своей зависимости от градиентов и получает вес каждого графика активации через его оценку прямого переноса на целевой класс. Конечный результат получается линейной комбинацией весов и графиков активации.

CAM также может быть применен ко многим аспектам:

- Например, он может обнаруживать полезные объекты в сцене.
- Найдите более абстрактные понятия в слабо обозначенных изображениях.
- Используя технологию визуализации CAM, я думаю, что CAM также может помочь нам сравнить модели и выбрать более подходящую структуру.

Мы также используем характеристики CAM, которые могут визуализировать объекты для сжатия изображений. Но на самом деле мы пришли к выводу: это не идеально.

Причина в том, что традиционно сети глубокой классификации обычно используют уникальные шаблоны определенных категорий для идентификации. Сгенерированная карта местоположения объекта может выделить только небольшую область целевого объекта, а не весь объект.

³Wang H, Wang Z, Du M, 2019. Score-CAM: Score-Weighted Visual Explanations for Convolutional Neural Networks. <https://arxiv.org/abs/1910.01279>

3.1.2. Состязательное дополнительное обучение

Поэтому я использовал новую модель: ACoL⁴ - Adversarial Complementary Learning, чтобы компенсировать эту слабость.

Предложена простая сетевая структура, которая содержит два параллельных классификатора для определения местоположения цели. При переходе вперед динамически находите некоторые отличительные целевые области во время классификации.

Это своего рода состязательное обучение. Два параллельных классификатора вынуждены использовать дополнительные целевые области для классификации и, наконец, совместно генерировать полное целевое позиционирование.

Предлагаемый ACoL включает в себя три компонента: Магистраль, Классификатор А и Классификатор В. Магистраль: полностью сверточная сеть, которая служит в качестве средства извлечения объектов. Карта объектов магистральной сети вводится в следующую параллельную классификационную ветвь. Две ветви содержат одинаковое количество сверточных слоев, за которыми следуют слой GAP и слой softmax для классификации.

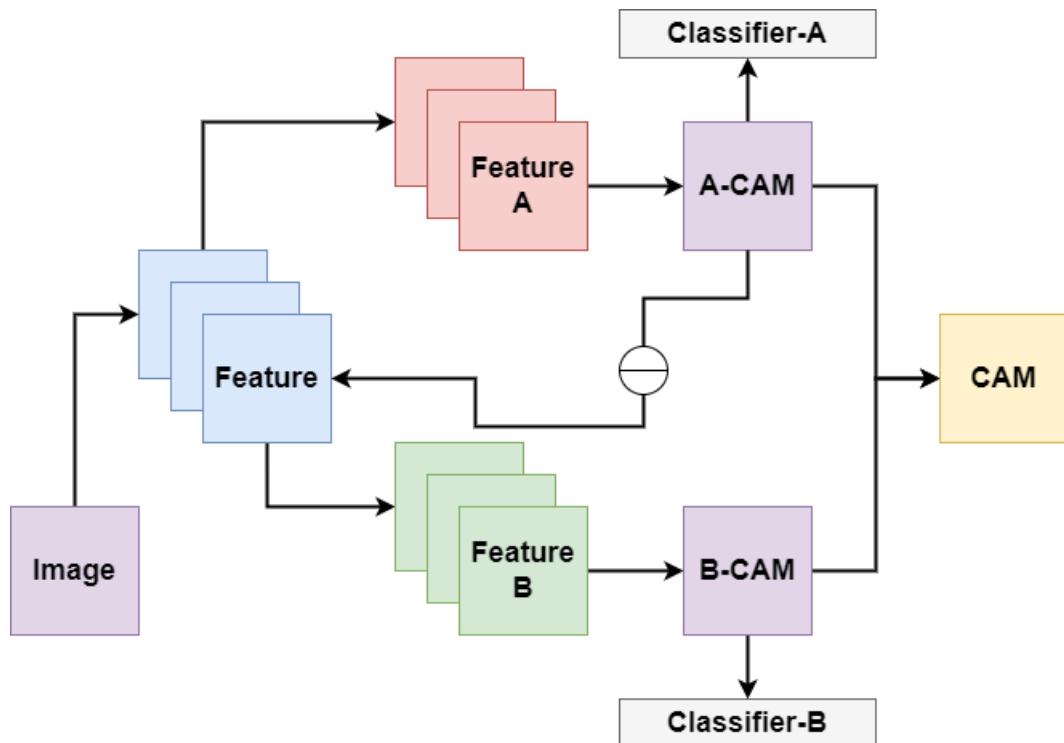


Рис. : Adversarial Complementary Learning

Карта позиционирования объекта предлагаемого способа для каждого изображения выше показывает классификатор А, классификатор В и объединенное изображение соответственно. Предлагаемые два классификатора могут обнаруживать разные части целевого объекта, тем самым обнаруживая всю область одной и той же категории на заданном изображении.

⁴Zhang X, Wei Y, Feng J, 2018. Adversarial Complementary Learning for Weakly Supervised Object Localization. <https://arxiv.org/abs/1804.06962>

В частности, входные характеристики классификатора В частично стираются под руководством различимой области, генерируемой классификатором А.

Выполните пороговую сегментацию на карте позиционирования классификатора А, чтобы определить различимые области. Затем соответствующая область на карте входных объектов классификатора В стирается с помощью 0 вместо этого для достижения конфронтации. Точнее, области, превышающие пороговое значение , стираются. Такая операция побуждает классификатор В использовать характеристики других областей целевого объекта для поддержки маркировки на уровне изображения. Наконец, карты позиционирования, сгенерированные двумя ветвями, объединяются для получения общей карты позиционирования целевого объекта. Используйте функцию Max для операции слияния. Весь процесс представляет собой сквозное обучение. Оба классификатора используют функцию потери перекрестной энтропии для обучения.

CAM может захватить только часть цели, в то время как ACoL может захватить большую часть цели.

3.1.3. Фильтрация объектов и классов

Теперь мы извлекли изображения одного класса. Однако методы реализации нескольких классов для каждой сети различны.

Пороговая фильтрация используется в MSROI и CAM. MSROI⁵ вычисляет оценку каждой категории, затем сохраняет вес, превышающий определенное пороговое значение, а затем суммирует каждую категорию для получения графика активации нескольких категорий. CAM сохраняет и включает в работу параметры первых нескольких типов после softmax, и полученный результат представляет собой диаграмму активации класса всех функций, необходимых для вычисления параметров в качестве области интереса.

Мы будем и дальше совершенствоваться на этой основе. Это хороший способ отфильтровать важные функции и вывести их вместе. Но некоторые ситуации не соответствуют логике ментального видения.

Представьте себе ситуацию, когда на изображении есть кошки и собаки, несколько цветов и куча игрушек на земле. Эта картина очень сложная, поэтому в ROI, полученном методом усечения порога, вероятно, будет отсутствовать некоторая информация. Например, некоторые объекты, которые очень важны, но занимают небольшую долю и им не хватает места, такие как цветы и игрушки, затем используйте эти объекты в качестве фона. В другом случае, если на картинке есть кошка и собака. У собак особенно особая текстура, но кошки чисто белые. Очевидно, что мы можем видеть характеристики этих двух категорий одновременно. Но если вы просто добавите их к простому взвешенному

⁵Prakash A, Moran N, Garber S, 2022. Semantic Perceptual Image Compression using Deep Convolution Networks. <https://arxiv.org/abs/1612.08712>

линейному суммированию, это сделает собаку, видимую на картинке, особенно четкой, но кошка имеет большую площадь и является пространственно избыточной.

Поэтому мы по-разному разобрались с операцией фильтрации и слияния нескольких категорий. Например, чтобы придать каждому типу коэффициента степень важности, мы можем рассматривать результат softmax последнего слоя полностью связанного слоя как важность этого класса. Конечно, это подходит только для сам или простых оценочных вычислений, которые особенно сложны и трудны для реализации при решении градиентов или дополнительном изучении объектов.

$$M^c = \text{softmax}(y^c) \cdot \left(\sum_c (\text{Relu}(w \cdot f_d^c(x, y) - \alpha) - \beta) \right)$$

На основе сам вычтите одну альфа-версию из каждой и выполните relu, чтобы удалить некоторые карты объектов низкой важности, а затем суммируйте и вычтите бета-версию, чтобы выполнить relu, чтобы удалить некоторые классы низкой важности. Цель этого шага - удалить некоторый шум. Конечно, вы можете установить эти два параметра обучения особенно малыми, потому что позже мы будем умножать на softmax, важность которого заключается в весе, что позволит разбавить некоторые типы функций и получить высокоточные многотипные диаграммы активации, которые сохраняются.

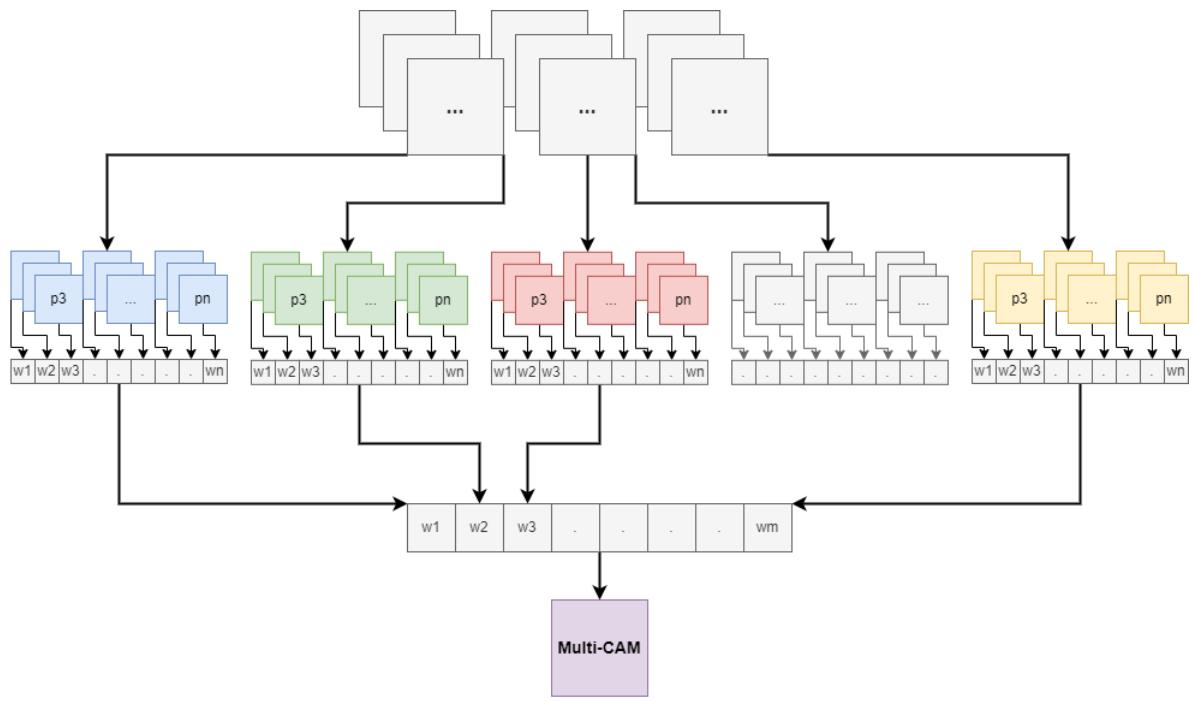


Рис. : Multi-class

Хотя это может привести к неудовлетворительным результатам оценки, поскольку цель сжатия изображения отличается от априорной информации изображения, также изменился подходящий для него метод сжатия. Это также яв-

ляется ограничением метода сжатия, основанного на психовизуальной избыточности, то есть он не является универсальным и не существует единого и абсолютно правильного метода сжатия. Однако на уровне объектов можно добиться сжатия, которое сортирует и умножает веса в соответствии с важностью объектов. В результате больше внимания уделяется различию различных областей объектов, и все это отражается на сжатом изображении.

3.2. Процесс сжатия

С помощью описанных выше операций мы получаем диаграмму активации класса, которую мы используем в качестве ROI для сжатия изображений. Существует несколько способов сжатия:

3.2.1. Линейное сжатие

P_{ij} - это оттенки серого для ROI пикселя (i, j) , поскольку ROI - это одноканальное изображение, что означает, что значение каждого пикселя представляется собой число от 0 до 255. Тогда Q_{ij} - это качество сжатия, рассчитанное на основе рентабельности инвестиций на данный момент, от Q_{start} до Q_{end} , например 30 и 70. Тогда формула для значения пикселя и качества сжатия представляет собой линейную функцию.

$$Q_{ij} = \frac{P_{ij} \cdot \text{range}(P)}{p_{max}} + Q_{start}$$

Затем мы сжимаем каждый пиксель изображения $Img_{original}$ в разной степени в зависимости от рассчитанного качества сжатия и, наконец, получаем сжатое изображение $Img_{compressed}$. Где f и g могут быть любым процессом сжатия и процессом декомпрессии, но должен быть параметр Q для качества сжатия операций с потерями. Таким образом, теоретически метод, описанный в этой статье, может быть объединен с любым уже существующим алгоритмом сжатия.

$$Img_{compressed} = \sum_i \sum_j f(Img_{original}^{ij}, Q_{ij})$$

$$Img_{original} = \sum_i \sum_j g(Img_{compressed}^{ij}, Q_{ij})$$

Вычисление каждого пикселя P приведет к увеличению объема вычислений по мере увеличения разрешения. Изображение $Img_{original}$ может быть сжато параллельно в соответствии с различными оттенками серого ROI, то есть Q_t делится на несколько дискретных значений от Q_{start} до Q_{end} , а затем каждое разрешение сжимается. То есть, следующая формула:

$$Img_{compressed} = \sum_t f(Img_{original}(Q_{ij} = Q_t), Q_t), \quad \forall Q_t \in [Q_{start}, Q_{end}]$$

3.2.2. Нелинейное сжатие

Но в этом расчете есть два недостатка:

- Непрерывное сжатие - это чрезмерная трата ресурсов и очень низкая практичность.
- При сжатии он основан на пространственной позиционной взаимосвязи между пикселями, чтобы устранить пространственную избыточность. Если вы сжимаете в соответствии с пикселями, позиционная взаимосвязь нарушается, и избыточность между пикселями может не быть устранена. Эффект сжатия небольшими партиями может быть не таким хорошим, как при сжатии всего изображения вместе.

Поэтому мы продолжаем совершенствовать метод, сжимаем изображение отдельно в соответствии с несколькими различными дискретными качествами сжатия, а затем в соответствии с результатами качества сжатия, определяемыми ROI, берем значения RGB пикселей из изображений, сжатых с различными качествами сжатия, а затем распараллеливаем сжатое изображение.

Пороговое значение квантиля

Хотя это несколько абстрактно, написать язык программирования очень просто и интуитивно понятно.

```
def q_cal(p):
    q = 50
    if p < q_20:      q = 20
    elif p < q_40:    q = 40
    elif p < q_60:    q = 60
    elif p < q_80:    q = 80
    else:             q = 100
    return q
```

Кусочные функции

Это формула качества сжатия для классификации с двумя разделениями, и можно задать качество сжатия объектов и фонов.

$$Q = \begin{cases} Q_{background} & \text{if } p < q_{threshold} \\ Q_{object} & \text{if } p \geq q_{threshold} \end{cases}$$

Или также выраженный на языке программирования:

```
def q_cal(p):
    if p < q_threshold: q = q_background
    else:                 q = q_object
    return q
```

Особенно полезным моментом является то, что если мы установим $Q_{object} = 100$ и $Q_{background} = 0$, мы можем получить усеченный вид объекта.

3.2.3. Энтропийные методы

Существует особый случай, когда нам нужно уделять больше внимания краевой области. Другими словами, нам нужно сохранить информацию о линии границы, но мы не обращаем внимания на объект и фон. Так что же мне делать в этой ситуации?

Я разработал алгоритм, который может вычислять энтропию на основе площади ROI, получать площадь края по пороговому значению и выполнять различные степени сжатия в соответствии с порядком энтропии. Или просто используйте его, чтобы компенсировать края предыдущих методов, вы можете использовать это как эффект “сглаживания”.

$$H(P) = E[I(P)] = E[\log_2 \frac{1}{P(p_i)}] = -P(p_i) \cdot \log_2 P(p_i), \quad \forall i = 1, 2, \dots n$$

Давайте сначала вычислим энтропию в матрице $n \times n$. Большинство алгоритмов сжатия, таких как jpeg, будут выполнять 8×8 блоков, поэтому мы вычисляем энтропию 8×8 . Вы можете установить различную точность в соответствии с вашими потребностями, например 4×4 и 16×16 . Затем уменьшите качество сжатия в соответствии с порядком энтропии.

$$Q = F_{cal}(H_{p_{8 \times 8}})$$

Используйте блоки с высокой энтропией, чтобы добавить дополнительный параметр, чтобы компенсировать недостаточное внимание, уделяемое краям при нелинейном сжатии.

$$Q = Q_t \oplus Q_{extra}$$

3.2.4. Прогрессивное взвешивание

Предыдущее сжатие может реализовать функцию сжатия исходного изображения в соответствии с ROI, но конечным средним Q нельзя управлять.

$$Q_{average} = N \sum_t Q_t \cdot n_t$$

Где N - общее количество пикселей на изображении, а n_t - количество различных Q. Мы обнаружили, что сжатие $Q_{average}$ всего изображения является апостериорным.

$$N = Count(P_{image})$$

$$n_t = Count(P(Q = Q_t))$$

И при сжатии я разработал следующий алгоритм: Вычислите сумму количества пикселей $N = h \cdot w$, то есть разрешение, а затем вычислите:

$$\frac{\sum_t q_t \cdot n_t}{N} = q_{given}$$

среди них n_t - это разрешение, при котором качество сжатия пикселей равно q_t , а затем в соответствии с формулой линейного суммирования в первом способ.

$$q_t = q_{start} + \frac{t \cdot \lambda_t \cdot (q_{end} - q_{start})}{N}$$

3.3. Применимость

Основной особенностью метода психовизуального сжатия является то, что он может быть наложен на любую сеть сжатия изображений. В начале этой главы я нарисовал блок-схему фреймворка сжатия, которая содержит процесс генерации ROI и процесс сжатия изображений. Фактически, метод, описанный в этой статье, можно комбинировать со многими существующими методами, такими как JPEG, BPG, Hide-and-Seek, ADL, Dropout, CNN, RNN и т.д. Я выбрал некоторые из этих существующих методов из многих существующих методов и объединил их с методами сжатия MS-ROI. Оказывается, что комбинированные эффекты очень хороши. Однако из-за проблем с пространством нет возможности показать все детали, поэтому сейчас я кратко покажу комбинированный метод и процесс, а также покажу эффект суперпозиции модели.

3.3.1. Для лучшего отображения ROI

Так называемое стирание заключается в удалении некоторой полезной и различающей информации. Есть надежда, что сеть сможет полагаться на оставшуюся информацию для получения точных прогнозов, что, естественно, улучшит способность сети извлекать более полные функции, а соответствующая CAM будет более точной. Я использовал метод ACoL в программе, который представляет собой способ использования прогностической информации для стирания, но, очевидно, есть много других вариантов. Вы можете выбрать подходящий метод улучшения CAM в соответствии с требованиями к сжатию, изображением, вычислительными ресурсами и т.д.

Простое разделение, стирание обычно включает в себя два метода:

1. Сотрите исходное изображение.
2. Стереть карту объектов.

Более подробная классификация районов:

1. Стирание просто: Стирание непосредственно на исходном изображении обычно выполняется случайным образом в соответствии с определенными правилами и обычно используется для улучшения данных.
2. Стирание с предсказанием: Использование информации, предсказанной сетью для стирания, аналогично обратному использованию механизма внимания. Например, сетевым вводом является исходное изображение, а камера реального времени используется для его стирания во время обучения, а затем выполняется расчет потерь. Этот метод обучения -> стирания -> переподготовки -> стирания часто называют состязательным стиранием.
3. Стирание с Dropout: Стратегии, подобные отсею, которые случайным образом стирают карты объектов в соответствии с определенными правилами, обычно не используют информацию сетевого прогнозирования, которая является средством регуляризации.

Hide-and-Seek/ Random Erasing/ Cutout/ Grid Mask

Эти четыре подхода по сути одинаковы, но есть некоторые различия в том, как они реализуются. Основная идея состоит в том, чтобы удалить некоторые области, чтобы CNN мог распознавать объекты только с помощью других областей, чтобы CNN мог использовать глобальную информацию изображения вместо локальной информации, состоящей всего из нескольких небольших объектов, для идентификации объектов. Кроме того, имитируя окклюзию, можно улучшить производительность и способность модели к обобщению.

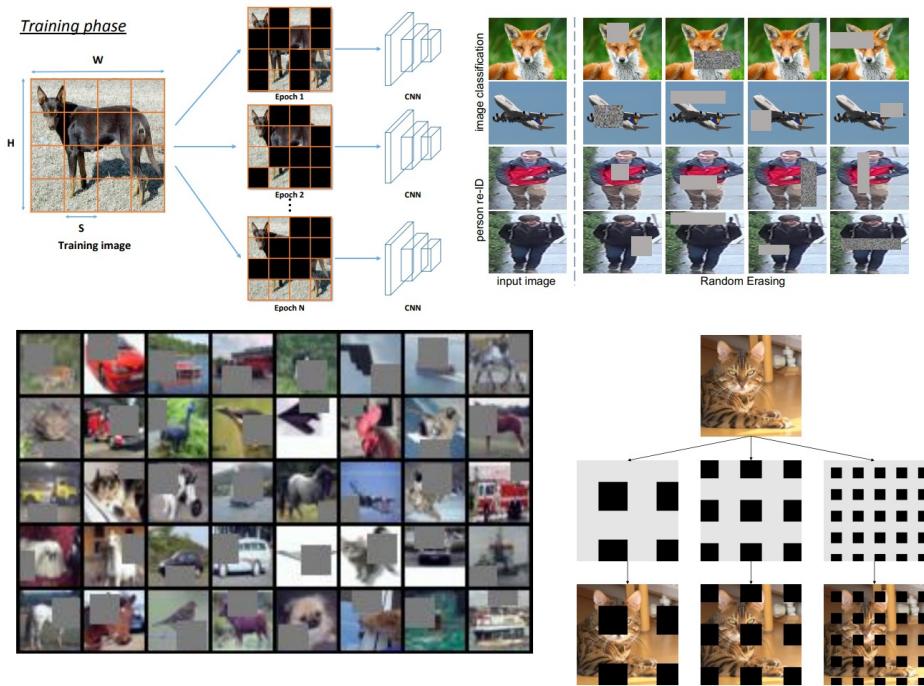


Рис. : Hide-and-Seek/Random Erasing/Cutout/Grid Mask

Так же, как и экспериментальные результаты в Hide-and-Seek, этот метод может значительно улучшить качество САМ и производительность сети.

AE-PSL

Сначала используйте исходное изображение для обучения сети классификации и используйте метод нисходящего внимания Attention, чтобы найти наиболее различимую область объекта на изображении. Затем извлеченная область стирается с исходного изображения, и удаленное изображение продолжает обучать другую классификационную сеть для определения местоположения других областей объектов. Повторяйте этот процесс до тех пор, пока сеть не станет хорошо сходиться на стертом обучающем изображении. Наконец, удаленная область объединяется как область извлеченного объекта.

SPG

Камера классификации все еще может различать передний план и фон. Хотя пиксели с высокой вероятностью переднего плана и фона могут не покрывать весь целевой объект и фон, они все равно дают важные подсказки. Исходя из этого, эти надежные начальные данные переднего плана и фона используются в качестве контроля, чтобы стимулировать восприятие объектов переднего плана и области сети, в которой распределен фон. Конкретный подход заключается в следующем: учитывая изображение, сначала создайте карты внимания на основе классификационной сети, а затем разделите карты внимания на три части: объект, фон и неопределенные области, на основе уровня достоверности

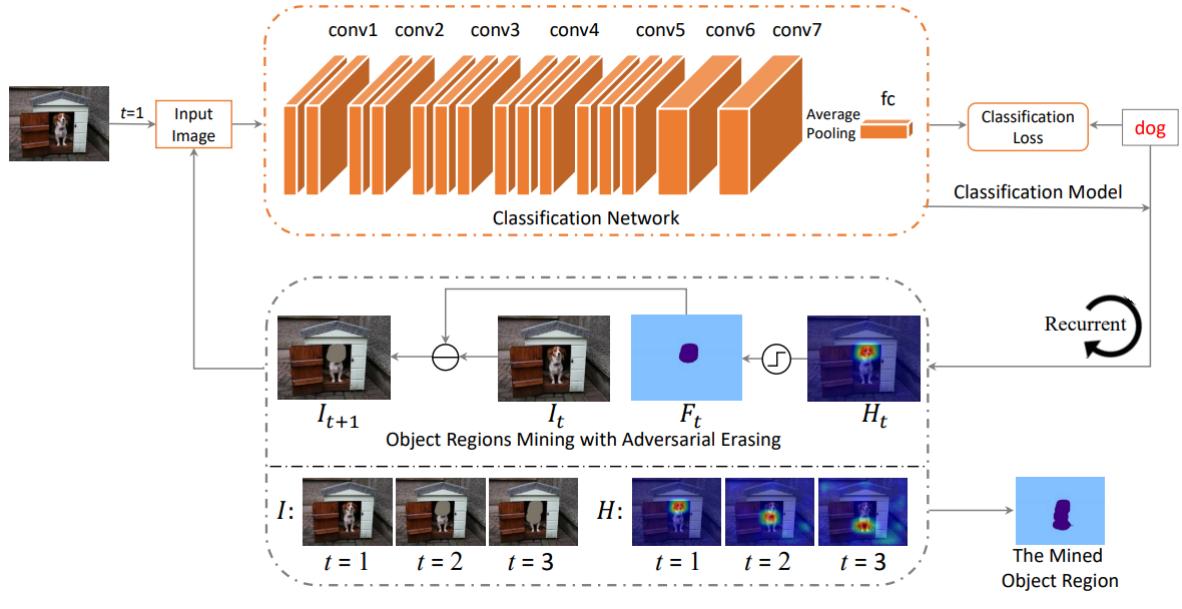


Рис. : AE-PSL

карт внимания.

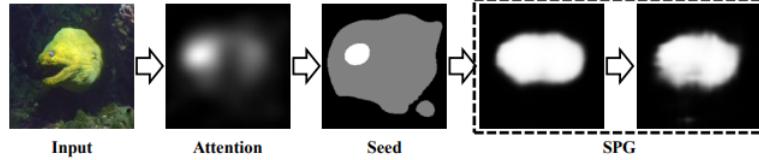


Рис. : SPG

Проблема с методом ACoL заключается в том, что он игнорирует корреляцию между пикселями. Поскольку похожие представления часто используются совместно между связанными пикселями, с помощью этих начальных значений можно найти некоторые надежные области переднего плана или фона. В этой статье используется механизм сверху вниз, использующий выходные данные верхнего уровня в качестве контроля нижнего уровня для получения информации о местоположении объекта. Маски SPG 0,1,255 производятся по следующей формуле:

$$M_{x,y} = \begin{cases} 0 & if \quad O_{x,y} < \delta_l, 0 < \delta_l < 1 \\ 1 & if \quad O_{x,y} > \delta_h, 0 < \delta_h < 1 \\ 255 & if \quad \delta_l \leq O_{x,y} \leq \delta_h, 0 < \delta_l < \delta_h < 1 \end{cases}$$

Рис. : SPG2

Этот метод особенно подходит для определенных сценариев и дает очень хорошие результаты при очень индивидуальных потребностях.

ADL

Идея относительно проста. Комбинируя пространственное внимание и операции стирания объектов, модуль вставляется на каждый слой свертки. Нет никаких параметров для изучения. Во время процесса обучения, каждый раз вперед, карта важности или выпадающая маска будут выбираться случайным образом для использования на карте объектов; во время процесса тестирования этот модуль использоваться не будет. Необходимо установить два суперпараметра: пороговое значение для получения маски удаления и вероятность случайного выбора; пороговое значение используется для управления только стиранием очевидных и легко различимых объектов. Случайно выбранный метод предотвращает стирание всех наиболее очевидных функций и приводит к путанице в сети.

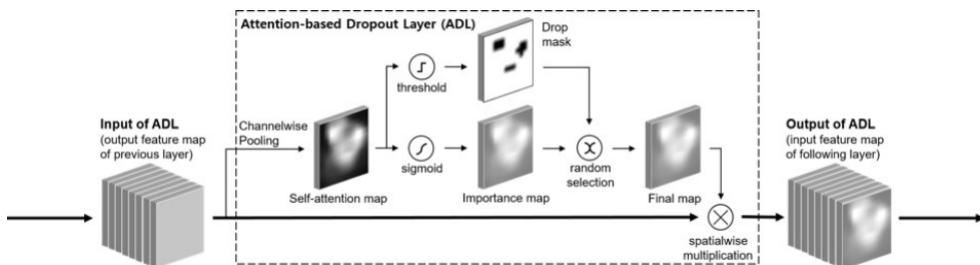


Рис. : ADL

Dropblock

На рисунке b показан случайный блок активации dropout, но после такого dropout сеть также получит ту же информацию из окрестности блока активации, куда упал dropout. Рисунок с представляет DropBlock. dropout часть прилегающей целой области, сеть будет уделять внимание изучению характеристик других частей собаки для достижения правильной классификации, тем самым демонстрируя лучшую способность к обобщению.

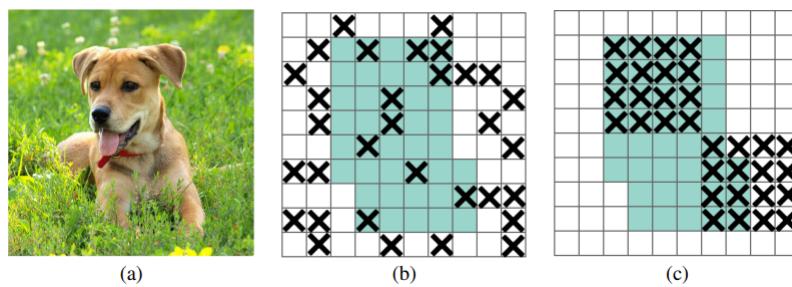


Рис. : Dropblock

FickleNet

Для предыдущего стирания на уровне изображения и стирания карты объектов обычно должен быть второй классификатор, а второй классификатор имеет только неоптимальную производительность. Кроме того, удаление объектов с возможностью распознавания приведет к путанице во втором классификаторе, и второй классификатор может быть неправильно обучен. Однако метод регионального выращивания будет в большей степени зависеть от качества исходных видов САМ, и его трудно выращивать в районах, которые не являются наиболее разборчивыми.

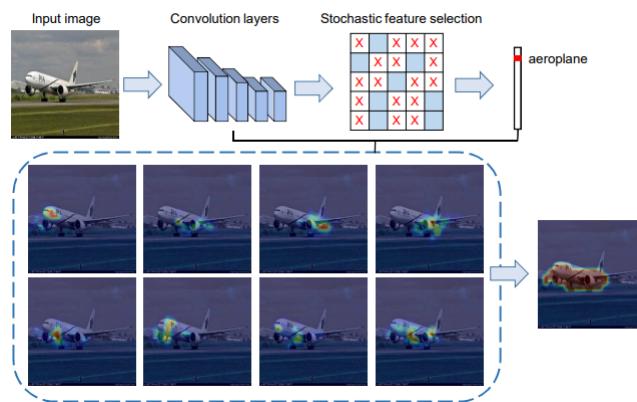


Рис. : FickleNet

FickleNet использует различные комбинации позиционирования на картах объектов CNN. Он случайным образом выбирает скрытые объекты, а затем использует их для получения оценок активации для классификации изображений. Согласованность каждого местоположения на карте объектов неявно изучается, в результате чего создается карта местоположения, которая может идентифицировать другие части объекта. Общий эффект достигается за счет единой сети путем выбора случайных пар скрытых объектов, что означает, что различные карты местоположения генерируются из одного изображения. Этот метод не требует каких-либо дополнительных шагов обучения, просто добавьте простой слой к стандартной сверточной нейронной сети. Выберите все скрытые объекты и активируйте эффект сглаживания в качестве фона и переднего плана вместе. Случайный выбор скрытых объектов может обеспечить более гибкую комбинацию, которая может более четко соответствовать части или фону объекта. Самое большое отличие от традиционного dropout заключается в том, что традиционный dropout используется только при обучении, но этот метод также будет использоваться при тестировании.

3.3.2. Для сжатия изображений

Мой алгоритм сжатия может быть объединен с любым традиционным алгоритмом сжатия и алгоритмом сжатия с глубоким обучением. Вот почему я

подробно описал непрерывную смену важных методов сжатия в истории во второй главе.

JPEG/JPEG2000/WebP/BPG

Этого нужно достичь только простым сжатием изображения в разной степени. Мы уже говорили об этом в предыдущем разделе, и мы не будем повторять их здесь.

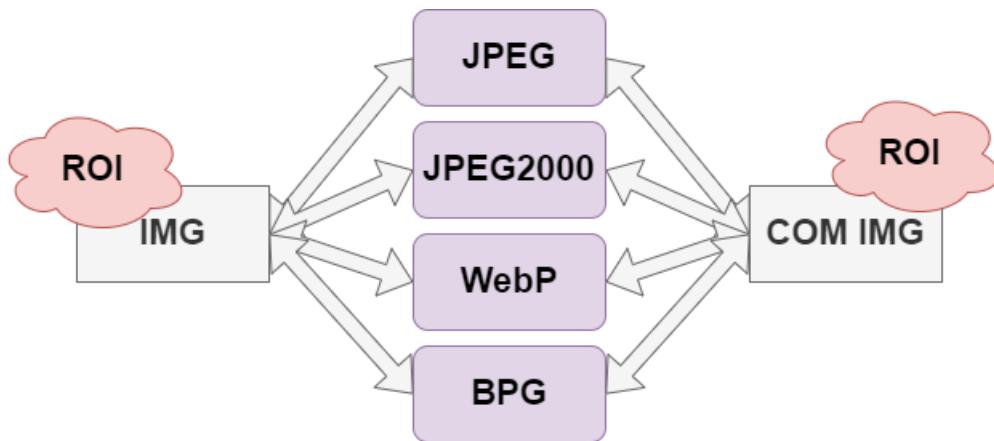


Рис. : JPEG/JPEG2000/WebP/BPG

CNN

Новый фреймворк сжатия, основанный на сверточных нейронных сетях, позволяет осуществлять высококачественное сжатие изображений. Эта структура состоит из двух частей: Com CNN⁶ используется для изучения оптимального компактного представления входного изображения и последующего кодирования изображения, а Rec CNN используется для восстановления высококачественного декодирующего изображения.

Сверточный блок CNN содержит возможность извлечения особенностей изображения. Итак, нужна ли вам CAM для помощи в извлечении и сжатии объектов? Это все еще необходимо изучить, чтобы получить выводы. Но, на мой взгляд, CNN, как кодировщику, больше не нужна CAM, чтобы помочь ему захватить интересующую область, что немного расточительно. Но мы все еще можем это сделать, что может снизить порог обучения CNN для сжатия изображений.

⁶Agustsson E, Tschannen M, Mentzer F, 2019. Generative Adversarial Networks for Extreme Learned Image Compression. <https://arxiv.org/abs/1804.02958>

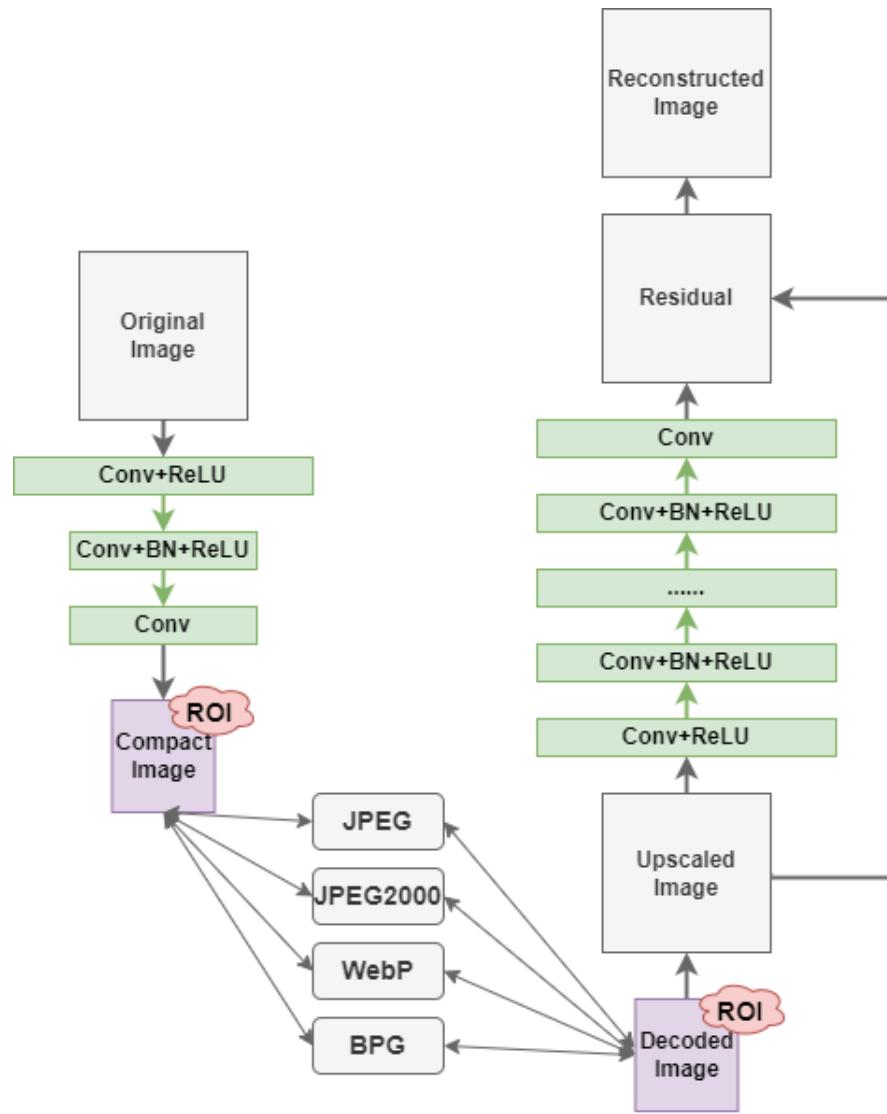


Рис. : MS-CAM для CNN

3.3.3. Для восстановления изображения

GAN

В сети "Extreme Learned"⁷ автор предлагает модель экстремального сжатия, основанную на обучении GAN, которая позволяет получать визуально удовлетворительные результаты при низких скоростях передачи данных. Существует два способа модели экстремального сжатия. Первый называется генеративным сжатием, а второй называется выборочным генеративным сжатием. Среди них выборочное сжатие требует использования диаграмм семантической сегментации. Мы можем использовать MS-CAM в качестве семантического графа, чтобы заменить практику в этой статье.

Выбор режима сжатия генерации происходит, когда зарезервирована определенная пользователем область с высокой детализацией, а семантическая кар-

⁷Jiang F, Tao W, Liu S, 2017. An End-to-End Compression Framework Based on Convolutional Neural Networks. <https://arxiv.org/abs/1708.00838>

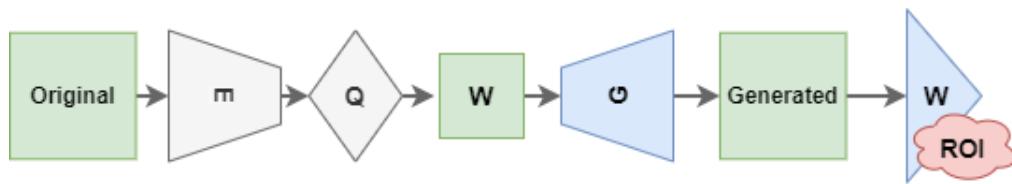


Рис. : MS-CAM для GAN

та используется для создания части изображения, которая не зарезервирована. Для модели SC автор создает одноканальную двоичную heatmap и пространственный размер квантованной карты объектов. Область, соответствующая 0, должна быть полностью синтезирована, в то время как область, соответствующая 1, сохраняет содержимое соответствующей области на карте объектов. Однако при сжатии область, требующая полного синтеза, имеет ту же семантику, что и исходное изображение. Предполагая, что семантические *s* хранятся отдельно, они проходят через feature extractor *F* перед подачей в генератор. Чтобы направлять сеть по семантике, автор выполняет операцию "маски" для искажения *d*, так что можно вычислить только потерю зарезервированной области. Более того, значение области, которая должна быть синтезирована в сжатой карте объектов *what*, установлено равным 0. Предполагая, что "тепловая карта" также сохранена, тогда необходимо закодировать только те области, которые необходимо зарезервировать в *what*, что может значительно уменьшить количество битов, которые необходимо сохранить. Обычно количество битов *what* намного больше, чем семантика хранилища и heatmap. Этот метод может значительно сэкономить количество битов.

3.3.4. Применение специальных сценариев

Я настроил несколько приложений для своего метода, которые работают в разных сценариях сжатия изображений. Вот краткое введение в предысторию, а примеры реализации будут показаны в следующей главе.

Мультикатегория

Сначала определите Мультикатегория: Мультикатегория относится к изображениям со сложным, абстрактным и без фона содержимым в изображении. Семантическая информация, необходимая для сжатия такого рода фотографий, может быть легко решена с помощью метода, описанного в этой статье.

- Сложные изображения: Сложность относится к присутствию нескольких объектов на изображении, и информация должна быть сохранена для каждого объекта.
- Абстрактное изображение: Объект не имеет определенного контура или нет очевидной границы между объектом и фоном. Например, морская вода, архитектура, пейзажи и т.д.



abstract image

Рис. : Multi

- Изображения с полным фоном: то есть на изображении нет фона или все они являются фоновыми, но все же существуют различия в степени важности, и необходимо различать разные степени важности.

Две категории

Две категории означает, что на изображении есть два типа объектов, но мы фокусируемся только на одном из них.



Рис. : Pets

Есть два способа решения такого рода проблем

1. Используйте несколько категорий, а затем выберите карту объектов фиксированного класса;
2. Используйте двустороннее классификационное обучение, а затем сгенерируйте диаграммы активации положительных и отрицательных выборок классов.

Безопасность

Сжатие защищенных изображений также относится к типу сжатия второго класса.

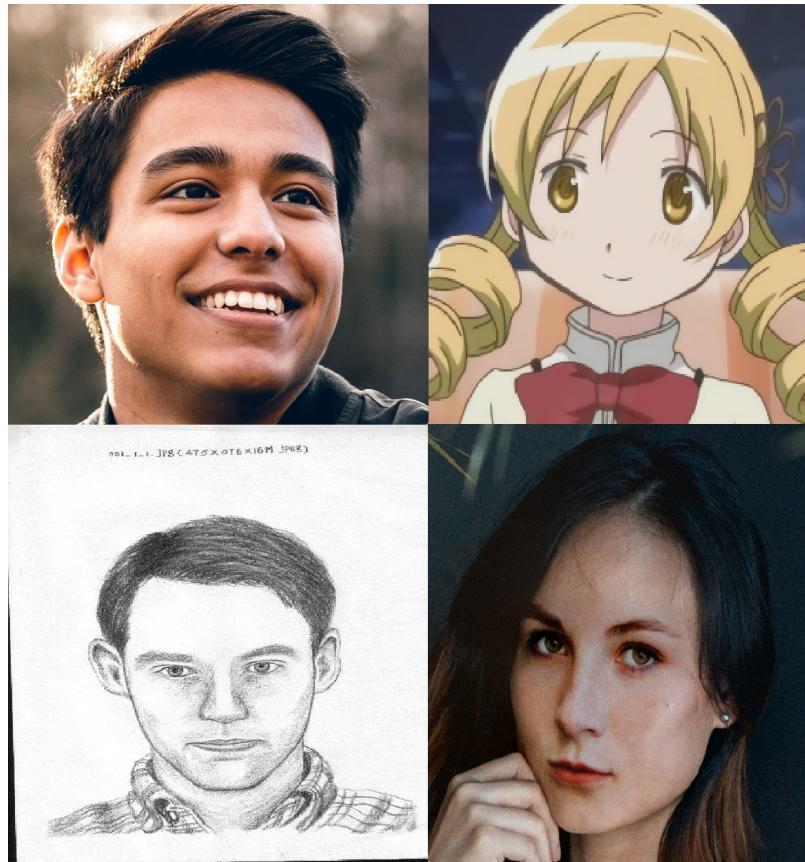


Рис. : Real or Fake

Преимущества этого двоякие

1. Может фильтровать ложную информацию, так что сжатие изображения имеет вспомогательные функции, такие как распознавание лиц и обнаружение безопасности;
2. Он отфильтровывает ложную информацию и экономит общее пространство. Это также способ добиться сжатия изображения.

Сжатие видео

Видео также может быть разделено на множество изображений, расположенных во времени, и мы также можем сжать каждое из этих изображений. В этой статье основное внимание уделяется только пространственной избыточности в изображениях. Здесь игнорируется временная избыточность и избыточность кодирования видео.



Рис. : Video Compression



Рис. : Video Compression 1



Рис. : Video Compression 2



Рис. : Video Compression 3

глава 4

Эксперименты

Основных целей эксперимента две: одна - проверить эффект сжатия метода, описанного в этой статье, а другая - протестировать приложение на основе нескольких сценариев, описанных в предыдущей главе. Для оценки эффекта сжатия используется метод контролируемого эксперимента, и индекс оценки изображений, сжатых в соответствии с интересующей областью, лучше, чем у изображений в контрольной группе. Затем мы протестируем три приложения: две категории, безопасность и сжатие видео. Применяемый метод оценки заключается в следующем: сначала внедрите эти функции и оцените, выполнимы ли они. Прежде всего, давайте представим несколько деталей реализации нашей программы: набор данных, метод оценки, некоторые инструменты и параметры.

4.1. Набор данных

Ниже приведен набор данных, используемый на этапе подготовки. Мы использовали ImageNet для обучения, потому что этот набор данных содержит большое количество и различные типы тегов, что означает, что он содержит характеристики большинства объектов. На самом деле нам не нужно знать, к каким категориям относятся эти характеристики. Мы только нужно изучить большинство характеристик. Этот набор данных соответствует моим требованиям. В качестве тестового набора для оценки используется Kodak PhotoCD. Это наиболее часто используемый набор данных. Он имеет высокое разрешение и его легко сравнить с другими моделями.

Ниже приведены наиболее часто используемые наборы данных для оценки сети классификации и сжатия изображений, и они также используются в качестве альтернатив на этапе отладки.

4.1.1. Обучающий набор

- ImageNet: Он содержит более 15 миллионов изображений, более 20 000 типов изображений и меток. Обычно он используется для таких задач, как классификация изображений и обнаружение целей. Благодаря большому объему данных изображений требуемый тип может быть выбран в реальных экспериментах.
- Cifar-100: Это цветное изображение размером 32 × 32, содержит 60 000 обучающих выборок и 100 категорий. Это классический набор данных для

классификации изображений и распознавания изображений. Разрешение изображения невелико, и категорий меньше.

- COCO: он включает в себя более 330 000 изображений в 80 категориях, из которых 200 000 изображений снабжены комментариями, которые в основном используются в областях распознавания целей, обнаружения объектов и сегментации объектов. Его единственная категория содержит большое количество данных изображений и богатые категории.

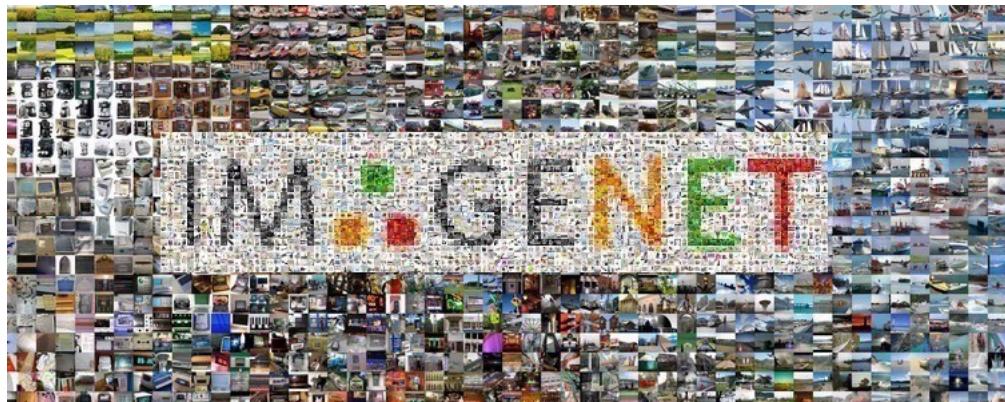


Рис. : ImageNet

4.1.2. Тестовый набор

- Kodak PhotoCD: в общей сложности 24 изображения с разрешением изображения 768 × 512 и разрешением около 400 000 пикселей. Это наиболее часто используемый набор тестовых данных для оценки сжатия изображений.
- CLIC: Как приложение, специально разработанное для сжатия изображений, оно обеспечивает более высокое разрешение изображений и фотографий, а разрешение снимков с профессиональной камеры составляет 1803 × 1175.
- Tecnick: около 1,4 миллиона пикселей, это также очень известный набор тестовых данных для оценки сжатия изображений.

4.2. Критерии оценки

Кратко познакомьтесь с инструментами и формулами, использованными при оценке. Инструмент оценки использует инструмент оценки видео нашей школы, а показатели оценки используют несколько часто используемых субъективных показателей оценки.

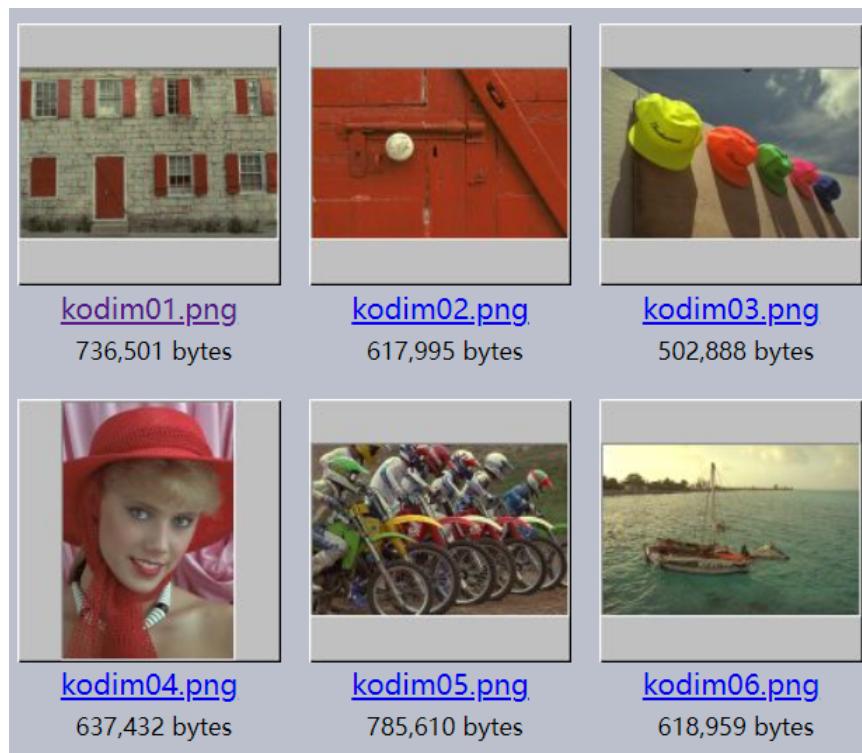


Рис. : Kodak PhotoCD

4.2.1. МГУ VQMT

МГУ VQMT - Video Quality Measurement Tool Инструмент измерения качества видео - это объективная программа оценки качества видео, разработанная Лабораторией графики и медиа Московского государственного университета. Он предоставляет множество методов оценки качества видео с полной ссылкой и методов оценки качества видео без ссылки, предоставляет функции расчета для нескольких основных показателей и визуальный интерфейс. Мы использовали три показателя для оценки PSNR, SSIM и MS-SSIM для оценки качества сжатия видео.

Но контент очень богатый, особенно для оценки видео. Нам не нужно так много. Дополнительные индикаторы и функции требуют платы, поэтому мы нашли более простой API EPFL VQMT.

4.2.2. EPFL VQMT

EPFL VQMT программное обеспечение обеспечивает быструю реализацию следующих объективных показателей:

- PSNR: Peak Signal-to-Noise Ratio
- SSIM: Structural Similarity
- MS-SSIM: Multi-Scale Structural Similarity
- VIFp: Visual Information Fidelity, pixel domain version

- PSNR-HVS: Peak Signal-to-Noise Ratio taking into account Contrast Sensitivity Function
- PSNR-HVS-M: Peak Signal-to-Noise Ratio taking into account Contrast Sensitivity Function and between-coefficient contrast masking of DCT basis functions

В этом программном обеспечении вышеуказанные показатели реализованы в OpenCV на основе оригинальных реализаций Matlab, предоставленных их разработчиками. Исходный код этого программного обеспечения может быть скомпилирован на любой платформе и требует только библиотеки OpenCV.

4.2.3. PSNR

Peak Signal-to-Noise Ratio - Пиковое отношение сигнала к шуму , называемое PSNR¹.

$$PSNR = 10 \cdot \log_{10} \frac{MAX_I^2}{MSE}$$

Среди них MAX_I - это максимальное значение, представляющее цвет точек изображения. Если каждая точка выборки представлена 8 битами, то она равна 255. Где I - несжатое исходное изображение, а K - сжатое изображение I.

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - K(i, j)]^2$$

Среди них MSE Mean Square Error - это среднеквадратичная ошибка.

4.2.4. SSIM

Structural similarity index measure² Индекс структурного сходства также является показателем, используемым для измерения сходства двух цифровых изображений. Учитывая два сигнала x и y, структурное сходство этих двух сигналов определяется как:

$$SSIM(x, y) = [l(x, y)]^\alpha [c(x, y)]^\beta [s(x, y)]^\gamma$$

Среди них $l(x, y)$ сравнивает яркость x и y, $c(x, y)$ сравнивает контрастность x и y, $s(x, y)$ сравнивает структуру x и y, а $\alpha > 0, \beta > 0, \gamma > 0$ - это параметр, который регулирует относительную важность яркости, контрастности и структуры.

¹Tanchenko A, 2014. Visual-PSNR measure of image quality.

²Wang Z, Bovik A, Sheikh H R, 2014. Image quality assessment: from error visibility to structural similarity.

$$l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}, \quad c(x, y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}, \quad s(x, y) = \frac{\sigma_{xy} + C_3}{\mu_x\mu_y + C_3},$$

$\mu_x, \sigma_x, \sigma_{xy}$ - это среднее значение, стандартное отклонение и ковариация, а c_1, c_2 и c_3 являются постоянными, которые используются для поддержания стабильности яркости, контрастности и структуры.

При фактическом использовании, для простоты, параметры обычно устанавливаются на $\alpha = \beta > 0 = \gamma = 0$ и $C_3 = \frac{C_2}{2}$:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

4.2.5. MS-SSIM

На основе алгоритма SSIM Multi-scale structural similarity³ for image quality assessment предлагается многомасштабный алгоритм оценки структурного сходства, как показано на рисунке, а именно алгоритм MS-SSIM.

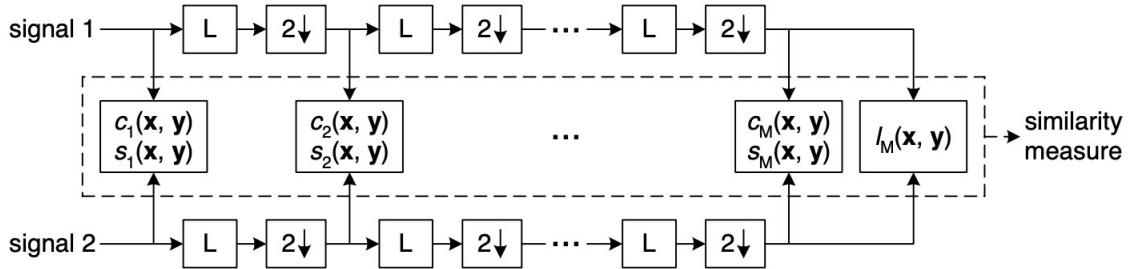


Рис. : MS-SSIM

L представляет фильтр низких частот, а 2 представляет понижающую дискретизацию с интервалом дискретизации 2.

Таким образом, MS-SSIM на самом деле является методом оценки качества изображения, который объединяет детали изображения с различными разрешениями. Для MS-SSIM $scale = 1$, а максимальный $scale = M$. Для шкалы со $scale = j$ сходство яркости, контрастности и структуры выражается как $l_j(X, Y), c_j(X, Y), s_j(X, Y)$ соответственно.

$$MS - SSIM(X, Y) = [l_M(X, Y)]^{\alpha_M} \cdot \prod_{j=1}^M [c_j(X, Y)]^{\beta_j} [s_j(X, Y)]^{\gamma_j}$$

В общем случае, пусть $\alpha_j = \beta_j = \gamma_j, j \in [1, M]$ мы получаем

³Wang Z, Simoncelli E P, Bovik A C 2003. Multiscale structural similarity for image quality assessment.

$$MS - SSIM(X, Y) = [l_M(X, Y)]^{\alpha_M} \cdot \prod_{j=1}^M [c_j(X, Y) \cdot s_j(X, Y)]^{\alpha_j}$$

4.3. Инструменты и параметры

4.3.1. Инструмент повышения эффективности

Потому что мои инновации - это не просто использование новых алгоритмов или алгоритмических оптимизаций. Различные сценарии и приложения также разрабатывались отдельно, и было проведено большое количество экспериментов. Поэтому очень важно снизить порог вычислительных ресурсов. Я использовал следующие два алгоритма, чтобы помочь в обучении сетевой модели: Трансферное обучение Ансамбль и методов.

Трансферное обучение

Нам приходится обучать большое количество сетей, чего невозможно достичь при фиксированном времени и ограниченных вычислительных ресурсах. Поэтому мы приняли трансферное обучение. Трансферное обучение уже использовалось при обучении сетей распознавания истинных и ложных лиц. В этой статье я хочу обучить большое количество повторяющихся сетей. В них есть только изменения в определенном слое или определенном параметре. Если я захочу начать с нуля отдельно, это займет слишком много времени и вычислительных ресурсов. Он очень подходит для такого рода экспериментов для обучения нескольких сетей САМ.

Одна вещь, которая особенно удачна, заключается в том, что мы можем перенести все обученные классификации в глубокие сверточные сети. Это очень важно. Нам даже не нужно проводить дополнительное обучение, существующая классификационная сеть уже очень конвергентна.

Ансамбль методов

Существует также метод повышения эффективности, который очень подходит для этой статьи - Ансамбль методов.

Существует два основных распространенных интегрированных алгоритма обучения: алгоритмы, основанные на Bagging, и алгоритмы, основанные на Boosting. Репрезентативные алгоритмы, основанные на Bagging, включают случайный лес, в то время как репрезентативные алгоритмы, основанные на Boosting, включают Adaboost, GBDT, XGBOOST и т.д.

Причина, по которой эта статья подходит, заключается в следующем: нам нужна не интеграция результатов, а интеграция САМ. Нет необходимости использовать пакетирование и повышение, но простая интеграция может быть до-

стигнута простым добавлением оттенков серого выходных данных ROI нескольких сетей.

4.3.2. Сетевая структура

Поскольку MS-CAM использует классификационную сеть с глубокой сверткой, это также означает, что можно использовать любую модель, такую как Vgg, Resnet, Vision Transformer и т.д. В ходе эксперимента результаты различных методов сравнивались не только с помощью контролируемых экспериментов, но и были протестированы различные классификационные сети, чтобы попытаться найти оптимальную и наиболее подходящую сеть.

VGG16 vs ResNet50 vs Vision Transformer

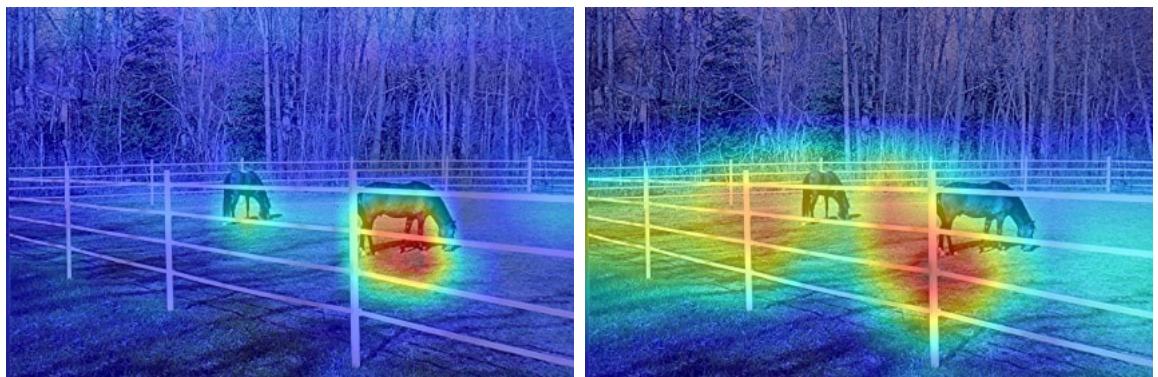


Рис. : GradCAM - VGG16 vs ResNet50

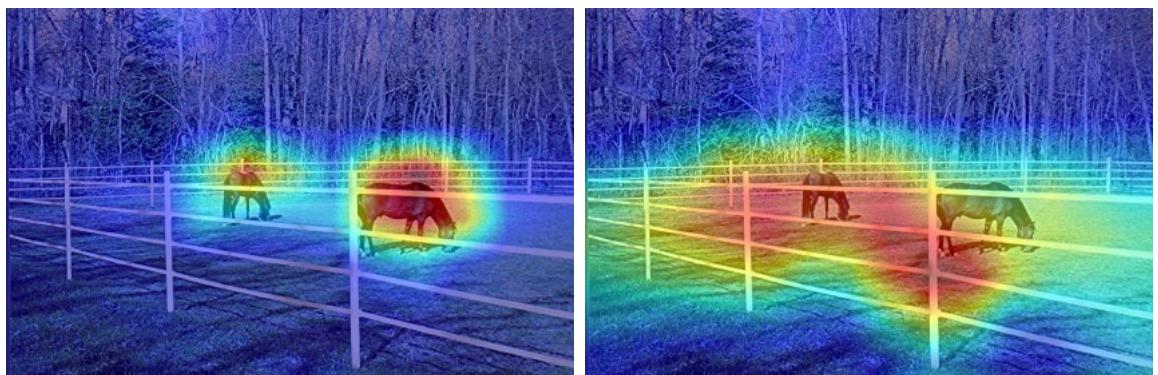


Рис. : ScoreCAM - VGG16 vs ResNet50

Параметры, которые я выбрал в соответствии с рекомендациями статьи:

```
def find_layer(model):
    if model.name == 'VGG16':
        target = model.features[-1]
    elif model.name == 'ResNet50':
        target = model.layer4[-1]
    elif model.name == 'ViT':
        target = model.blocks[-1].norm1
    return target
```

На следующих рисунках показаны только наиболее реалистичные результаты ROI, игнорируя некоторые методы сглаживания: aug smooth и eigen smooth.

```
def __init__():
    aug_smooth = False
    eigen_smooth = False
```

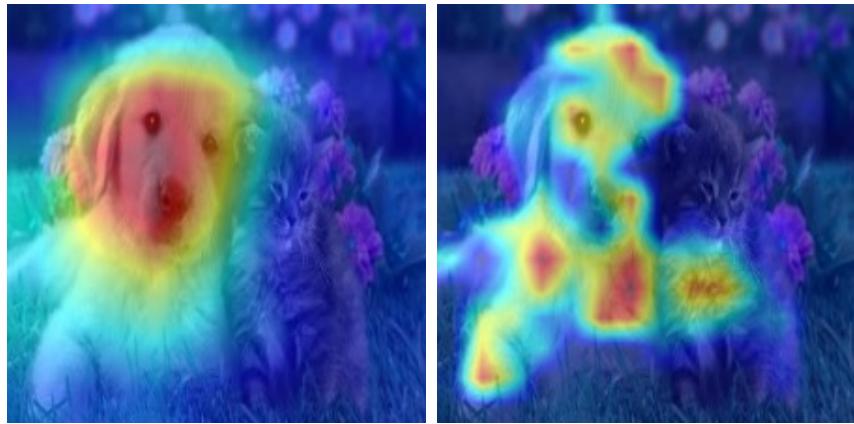


Рис. : GradCAM - ResNet50 vs ViT



Рис. : ScoreCAM - ResNet50 vs ViT

4.4. Результаты

В этом разделе мы сначала сравним результаты данных многотипного сжатия изображений нескольких моделей, а затем покажем четыре примера приложений.

Среди них моя модель изменена на основе JPEG и CAM, поэтому мы напрямую сравниваем MSROI в качестве базовой линии, чтобы доказать, что наш индекс субъективной оценки немного выше, чем MSROI. Затем мы внедрили четыре приложения, каждое из которых имеет разные методы реализации. "Мультикатегория" проблем были доказаны результатами контролируемых испытаний. Проблема "Две категории" заключается в том, что это функция с сильными субъективными требованиями, поэтому мы показываем только картинки. Его преимущества уже были продемонстрированы в процессе реализации. Что

касается вопроса "безопасности изображения мы сравним сжатие реальной информации и ложной информации, чтобы оценить полезность и преимущества этого. Что касается сжатия "видео я сохранил видео с черным фоном и белыми объектами. По сравнению со сжатием цветного видео это экономит разрыв в десятки раз. Здесь теряется цвет, априорная информация и т.д., И сохраняется только информация о местоположении видеообъектов, так что это неизбежно что степень сжатия особенно высока.

4.4.1. Мультикатегория

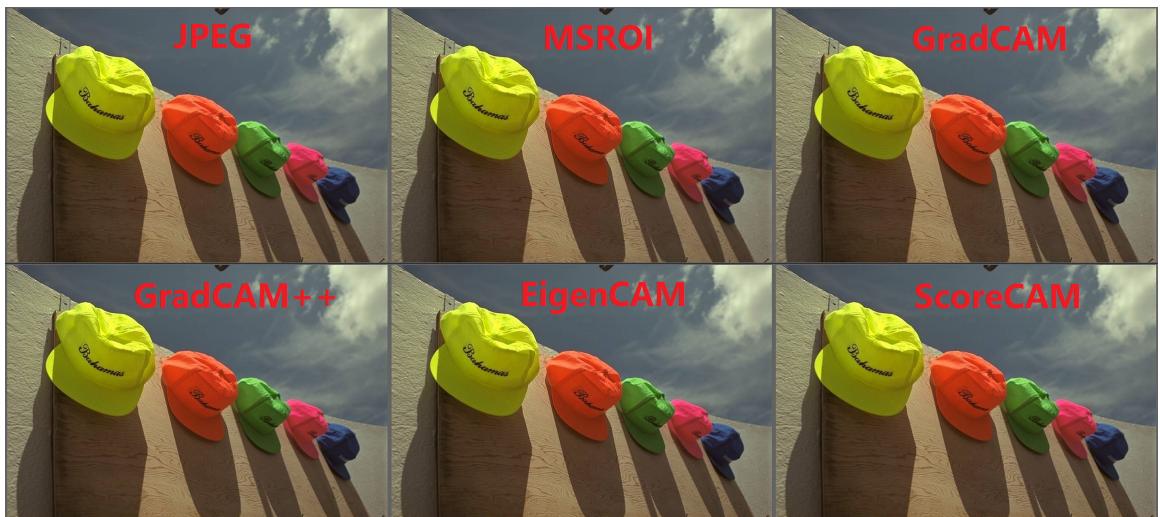


Рис. : Image Compression

Я протестировал эффекты сжатия изображения при трех степенях сжатия по отдельности. Все они эквивалентны качеству сжатия $Q = 50, 25, 75$. Результирующее изображение имеет глубину цвета $BPP = 0.075, 0.050, 0.114$. Их степени сжатия сосредоточены около $R = 25.5, 17.0, 11.2$. GradCAM лучше работает при низких степенях сжатия, в то время как EigenCAM и ScoreCAM лучше работают при высоких степенях сжатия.

Таблица : Результаты Q:50 BPP:0.075

Метод	Ratio	MSE	PSNR	SSIM	MS-SSIM
MSROI	16.69	0.000343	37.5464	0.9353	0.9042
GradCAM	17.00	0.000342	37.5639	0.9370	0.9036
GradCAM++	16.92	0.000371	37.5652	0.9370	0.9035
EigenCAM	16.85	0.000352	37.4470	0.9366	0.9034
ScoreCAM	16.83	0.000356	37.4776	0.9364	0.9027

Таблица : Результаты Q:25 BPP:0.050

Метод	Ratio	MSE	PSNR	SSIM	MS-SSIM
MSROI	25.40	0.000140	40.7228	0.9727	0.9065
GradCAM	25.57	0.000140	40.5692	0.9728	0.9011
GradCAM++	25.50	0.000139	40.6999	0.9735	0.9010
EigenCAM	25.67	0.000204	38.9895	0.9640	0.8799
ScoreCAM	25.69	0.000204	39.0253	0.9654	0.8814

Таблица : Результаты Q:75 BPP:0.114

Метод	Ratio	MSE	PSNR	SSIM	MS-SSIM
MSROI	11.19	0.000390	37.1160	0.9322	0.8703
GradCAM	11.15	0.000388	37.1434	0.9311	0.8707
GradCAM++	11.12	0.000389	37.1267	0.9310	0.8701
EigenCAM	11.05	0.000339	37.6364	0.9367	0.8757
ScoreCAM	11.13	0.000340	37.6248	0.9364	0.8749



Рис. : Image Compression 2

4.4.2. Две категории

Это не для того, чтобы полностью очертить объект. Если это так, мы можем использовать обнаружение объектов и семантическую сегментацию. Хорошая сеть классификации обучается для получения карт объектов, которые обеспечивают более высокое качество сжатия характеристик этих различных объектов, так что существует высокая вероятность того, что искусственно оцененные объекты будут сохранены. Ниже приведены два набора изображений: один - кошки и собаки, а другой - найти части всех изображений животных, на которых изображены коровы.



Рис. : Pets

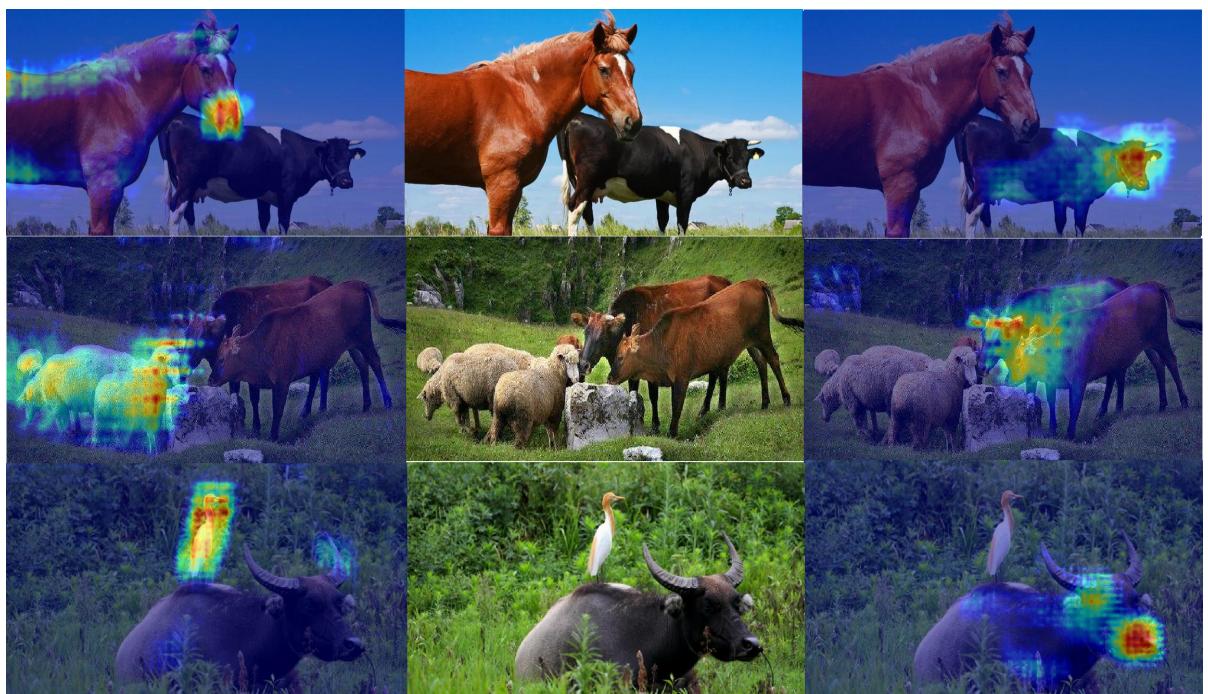


Рис. : Pets2

Две категории иногда являются просто еще одним способом выражения особых потребностей. Часто у нас возникают такие потребности: нам нужен альбом для сохранения домашних животных или альбом строительных блоков, сделанных нами самими, но когда мы отправляем их на облачный сервер, серверу требуется сжатие изображений, потому что у них не так много места. Если мы сожмем их таким же образом, возможный результат будет следующим: домашние животные не будут четкими, а строительные блоки не будут четкими. Если есть целенаправленное сжатие объектов, мы можем получить изображение нужного нам объекта, а не всего изображения.

4.4.3. Безопасность

Многие люди изучают область безопасности, и многие люди изучают визуальное сжатие. Но на самом деле их можно хорошо комбинировать, что будет беспрогрызным подходом. Мы можем реализовать фильтрацию ложной информации во время процесса сжатия и сохранить информацию о безопасности с небольшим объемом данных. Мой пример здесь таков: При истинном и ложном распознавании лиц сохраняется реальное лицо, а помехи от ложного лица устраняются на основе истинного и ложного распознавания.

Повторите вопрос: надеемся, что можно отделить набор данных, смешивающих настоящее лицо с искусственным лицом. Но как отделить правильное лицо из набора данных? Очевидно, что это проблема с классификацией, которая контролируется. Затем мы рассмотрим, как сохранить эти данные. Очевидно, нам не нужна эта ложная информация. Мы можем использовать высокие оттенки серого для замены поддельных лиц и фона, чтобы сохранить информацию, которая нам действительно нужна в будущем.

4.4.4. Сжатие видео

Повторите вопрос: надеемся, что можно отделить набор данных, смешивающих настоящее лицо с искусственным лицом. Но как отделить правильное лицо из набора данных? Очевидно, что это проблема с классификацией, которая контролируется. Затем мы рассмотрим, как сохранить эти данные. Очевидно, нам не нужна эта ложная информация. Мы можем использовать высокие оттенки серого для замены поддельных лиц и фона, чтобы сохранить информацию, которая нам действительно нужна в будущем.

Этот метод может быть использован для сохранения видеозаписи траектории движения любого объекта. Конечно, предпосылка заключается в том, что нам не нужна априорная информация об объекте, а нужно только обратить внимание на его траекторию. Сохраненное видео также может быть использовано в качестве регрессии для прогнозирования следующего хода. Это также направление, которое мы можем изучить на следующем шаге.

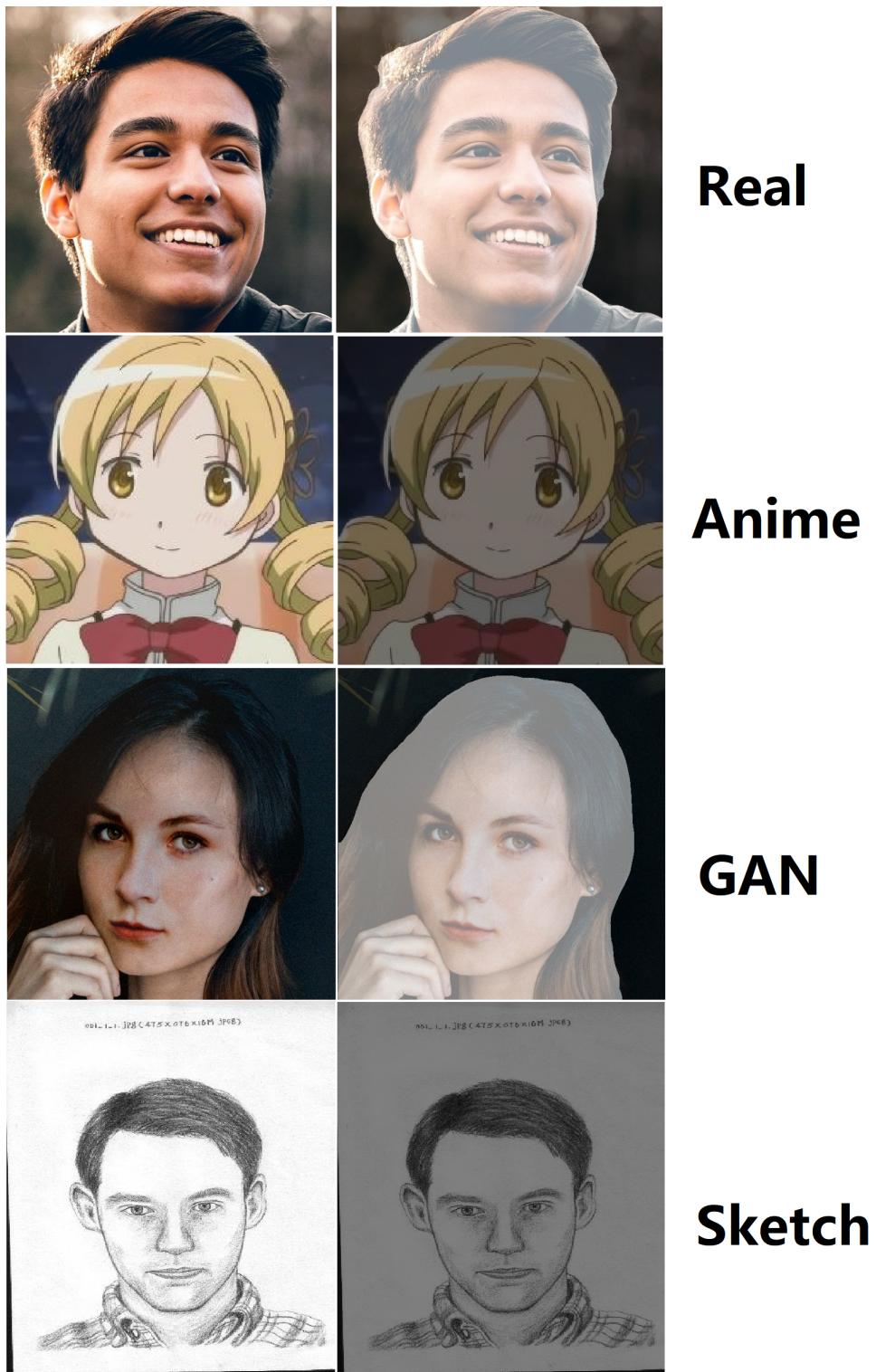


Рис. : Real Fakes

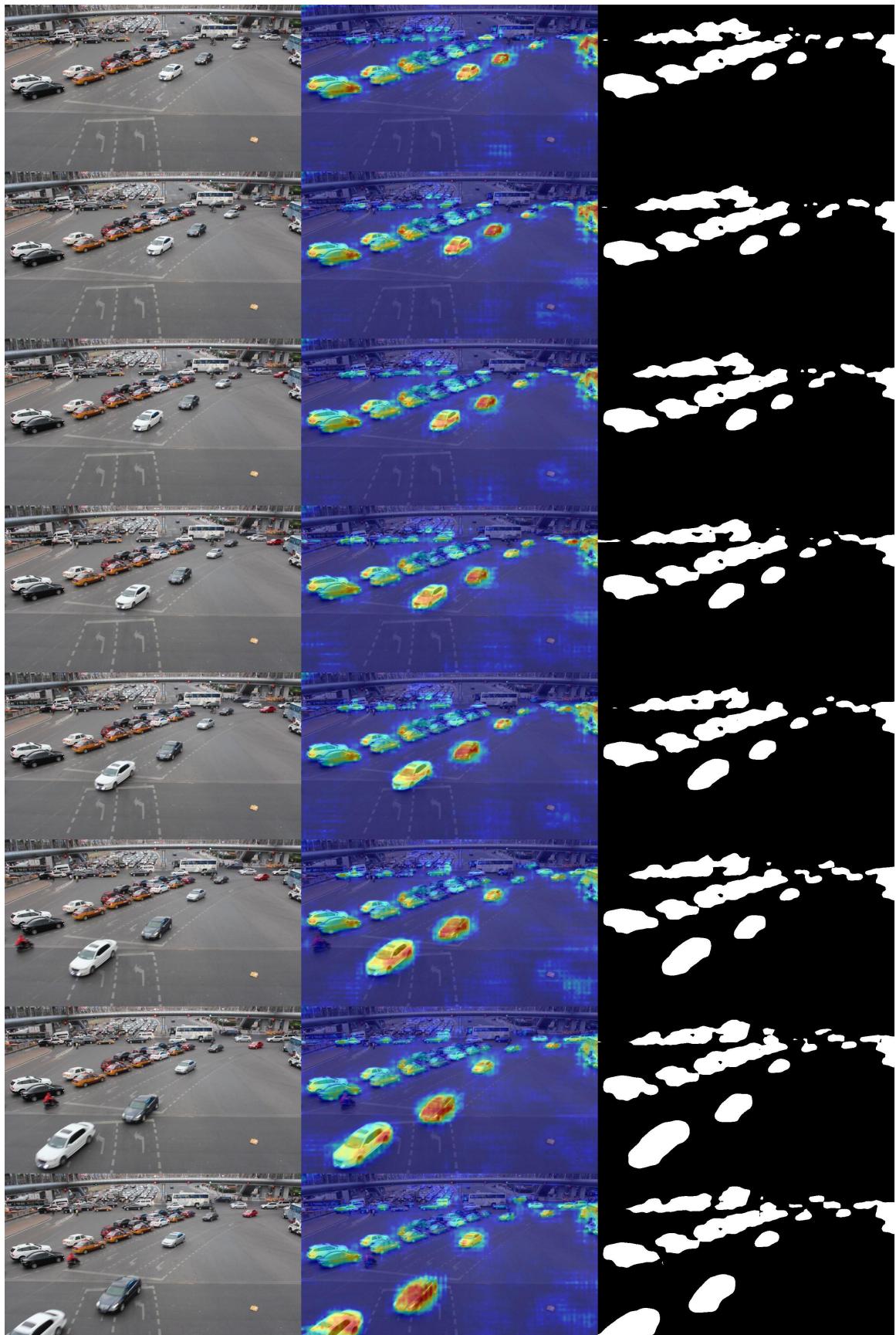


Рис. : Video Compression

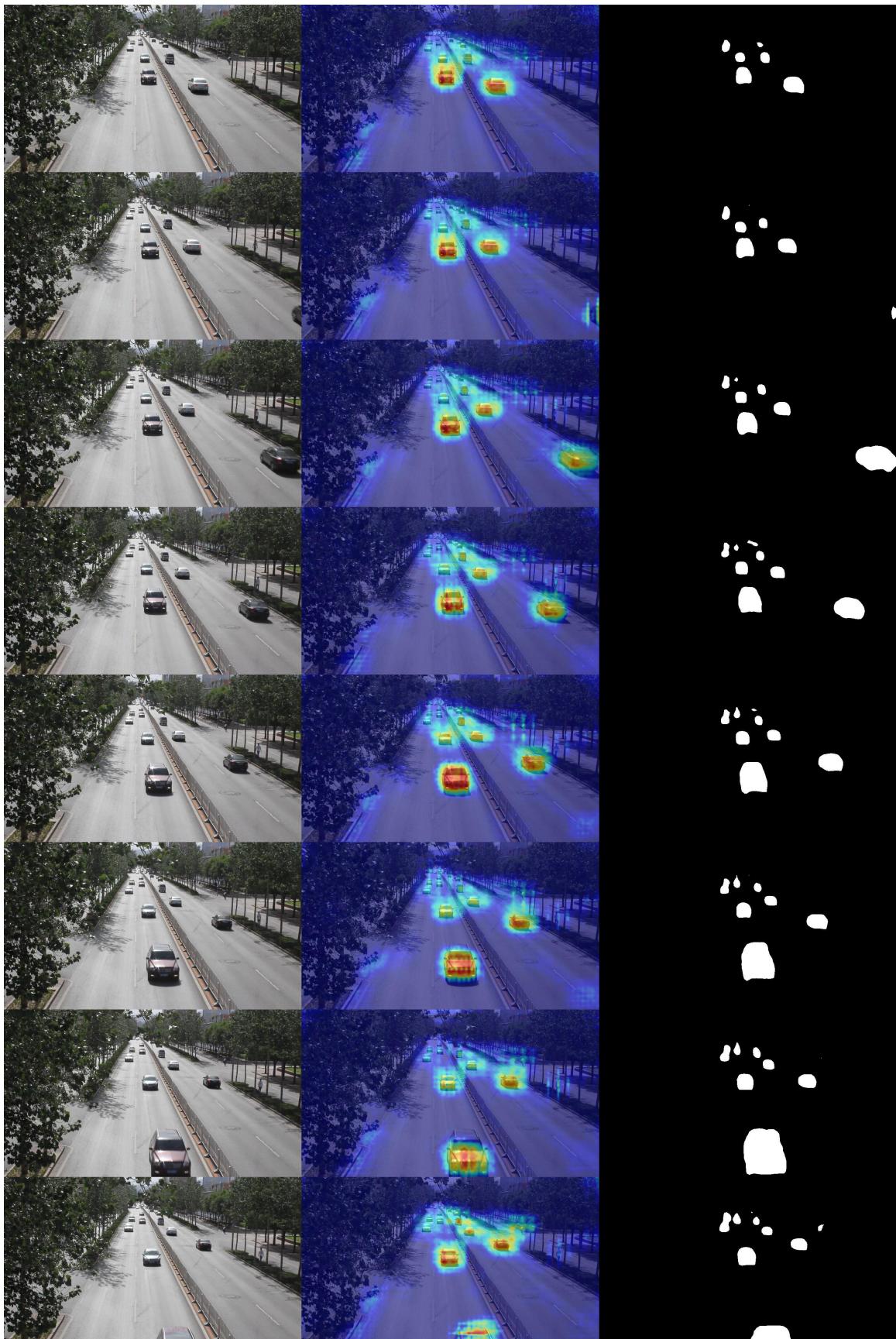


Рис. : Video Compression2

глава 5

Заключение

5.1. Вывод

Традиционный метод сжатия изображений и метод глубокого обучения deep learning быстро развиваются. Однако в этой статье используется уникальный подход, использующий новый метод, основанный на психовизуальной перспективе изображений, и это также еще один уникальный способ сжатия изображений нейронными сетями. Этот метод может быть связан с любой нейронной сетью, используемой для сжатия изображений, и мы убедились, что он превосходит некоторые существующие модели. Самое главное, что мы перечислили и внедрили четыре группы приложений. Для этих приложений уже существует множество решений, которые следует рассмотреть и решить с точки зрения сжатия изображений.

Но у нас все еще есть много недостатков

- Сравнение с обнаружением цели и семантической сегментацией не подтвердилось.
- Соединение с некоторыми методами сжатия - это всего лишь простая реализация, но вопрос о том, является ли эффект удовлетворительным, все еще нуждается в дальнейшем подтверждении.
- Можно ли лучше оптимизировать практику, используемую для сжатия видео для устранения пространственной избыточности?
- Разработка приложений для восстановления изображений не ведется.

5.2. Направление исследований

Ввиду этих нерешенных проблем я перечислил некоторые направления, в которых я могу продолжить свои исследования.

- Сравнительный эксперимент с обнаружением цели и семантической сегментацией.
- Для применения семантики в GAN ROI используется для дискриминатора D, чтобы способствовать лучшему обучению генератора G.
- Оптимизируйте результаты CAM с помощью "устранения".
- Можно ли дополнительно комбинировать сжатие видео в сочетании с традиционными методами или методами глубокого обучения с методами, описанными в этой статье?
- Симметричное отображение этапа сжатия изображения и его применение при реконструкции изображения.

Спасибо!

глава 6

Список

Таблица : Framework

метод	ссылка
CAM	https://github.com/jacobgil/pytorch-grad-cam
MSROI	https://github.com/iamaaditya/image-compression-cnn
ACoL	https://github.com/xiaomengyc/ACoL
Com/Rec CNN	https://github.com/compression-framework
Extreme Learned	https://github.com/Justin-Tan/generative-compression

Таблица : Compression

метод	ссылка
JPEG	https://pillow.readthedocs.io/en/stable/reference/Image.html
JPEG2000	https://glymur.readthedocs.io/en/latest/how_do_i.html
WebP	https://pypi.org/project/webp/
BPG	https://bellard.org/bpg/

Таблица : VQMT

метод	ссылка
MSU VQMT	https://www.compression.ru/
EPFL VQMT	https://github.com/Rolinh/VQMT

Таблица : Data sets

метод	ссылка
ImageNet	https://image-net.org/download.php
Cifar-100	https://www.cs.toronto.edu/~kriz/cifar.html
COCO	https://cocodataset.org/download
Kodak PhotoCD	http://www.cs.albany.edu/~xypan/research/snr/Kodak.html
CLIC	https://compression.cc/
Tecnick	https://testimages.org/

Литература

- [1] Wallace G K, 1991. The JPEG still picture compression standard.
- [2] Rabbani MJoshi R, 2002. An overview of the JPEG 2000 still image compression standard.
- [3] Web. WebP Image format. <https://developers.google.com/speed/webp/>
- [4] Fabrice Bellard. BPG Image format. <https://bellard.org/bpg/>
- [5] Liu H, Chen T, Shen Q, 2018. Deep Image Compression via End-to-End Learning. <https://arxiv.org/abs/1806.01496>
- [6] Toderici G, Vincent D, Johnston N, 2017. Full Resolution Image Compression with Recurrent Neural Networks. <https://arxiv.org/abs/1608.05148>
- [7] Tschannen M, Agustsson E, Lucic M, 2018. Deep Generative Models for Distribution-Preserving Lossy Compression. <https://arxiv.org/abs/1805.11057>
- [8] Zhou B, Khosla A, Lapedriza A, 2015. Learning Deep Features for Discriminative Localization. <https://arxiv.org/abs/1512.04150>
- [9] Selvaraju R, Cogswell M, Das A, 2016. Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. <https://arxiv.org/abs/1610.02391>
- [10] Wang H, Wang Z, Du M, 2019. Score-CAM: Score-Weighted Visual Explanations for Convolutional Neural Networks. <https://arxiv.org/abs/1910.01279>
- [11] Zhang X, Wei Y, Feng J, 2018. Adversarial Complementary Learning for Weakly Supervised Object Localization. <https://arxiv.org/abs/1804.06962>
- [12] Prakash A, Moran N, Garber S, 2022. Semantic Perceptual Image Compression using Deep Convolution Networks. <https://arxiv.org/abs/1612.08712>
- [13] Agustsson E, Tschannen M, Mentzer F, 2019. Generative Adversarial Networks for Extreme Learned Image Compression. <https://arxiv.org/abs/1804.02958>
- [14] Jiang F, Tao W, Liu S, 2017. An End-to-End Compression Framework Based on Convolutional Neural Networks. <https://arxiv.org/abs/1708.00838>
- [15] Tanchenko A, 2014. Visual-PSNR measure of image quality.
- [16] Wang Z, Bovik A, Sheikh H R, 2014. Image quality assessment: from error visibility to structural similarity.
- [17] Wang Z, Simoncelli E P, Bovik A C 2003. Multiscale structural similarity for image quality assessment.