

СОЗДАНИЕ ОБУЧАЮЩЕЙ ВЫБОРКИ ДЛЯ СЕГМЕНТАЦИИ И ДЕТЕКЦИИ ОБЪЕКТОВ С ПОМОЩЬЮ ГЕНЕРАЦИИ СИНТЕТИЧЕСКИХ ИЗОБРАЖЕНИЙ МЕТОДАМИ НЕЙРОННЫХ СЕТЕЙ

И.М. Орлова

Московский авиационный институт (национальный исследовательский университет)
Россия, 125993, Москва, Волоколамское шоссе, 4
E-mail: iraorlova2019@gmail.com

Ключевые слова: семантическая сегментация, детекция объектов, нейронные сети, обучающая выборка, генерация изображений, стилизация изображений.

Аннотация: Данная работа посвящена разработке программного обеспечения, реализующего создание обучающей выборки для нейронных сетей сегментации и детекции объектов. Приводится описание API Fusion Brain, используемого для генерации изображений. Рассматриваются алгоритмы сегментации, позволяющие произвести семантическую разметку данных. Приводится алгоритм переноса стиля (Neural Style Transfer) для улучшения качества обучения нейронных сетей.

1. Введение

Одной из главных проблем, с которой сталкиваются разработчики проектов машинного обучения, является необходимость в ручной разметке обучающих данных, что требует значительных затрат по времени и усилиям, особенно при работе с большими объемами информации, а также может привести к плохому обучению нейронной сети из-за недостаточного количества и качества данных.

В данной работе рассматривается проблема создания обучающего набора данных для нейронных сетей сегментации и детекции объектов. Например, в работах [1, 2] было размечено небольшое количество данных вручную, что привело к плохому качеству выделения нейронной сетью сегментационной маски объектов.

Решением данной проблемы является автоматизация процесса подготовки обучающей выборки, включающая в себя следующие этапы:

- генерация не зашумленных изображений с помощью нейронной сети;
- выделение масок или ограничивающих рамок объектов с помощью алгоритмов сегментации;
- стилизация изображений для улучшения качества сегментации изображений определённого стиля.

2. Генерация изображений API Fusion Brain

Для генерации не зашумленных изображений в работе использовалось API Fusion Brain [3], которое предоставляет доступ к модели искусственного интеллекта Kandinsky 3.0, разработанной командой Sber AI при партнёрской поддержке учёных из Института

искусственного интеллекта AIRI на объединённых датасетах Sber AI и компании SberDevices [4]. Взаимодействие с API осуществлялось с помощью библиотеки http-запросов Requests на языке программирования Python.

Kandinsky 3.0 представляет собой диффузионную модель для генерации высококачественных изображений по текстовому описанию (англ. Text-to-image). Общий принцип работы диффузионных моделей состоит из двух этапов: процесса прямой и обратной диффузии. Процесс прямой диффузии заключается в итеративном добавлении шума к исходному изображению. Процесс обратной диффузии заключается в итеративном шумоподавлении и создании нового изображения, похожего на исходное, с помощью обученной нейронной сети. Для этой цели часто используется архитектура U-Net. При генерации изображений в диффузионных моделях Text-to-image в качестве дополнительных признаков в процессе обучения передаются закодированные с помощью некоторой языковой модели части текста [5].

Пример генерации изображения с помощью API Fusion Brain для запроса «Красное яблоко на белом фоне» приведён на рис. 1.



Рис. 1. Пример генерации изображения.

3. Алгоритмы сегментации

Для создания масок и ограничивающих рамок (англ. Bounding boxes) объектов были применены реализации алгоритмов сегментации из библиотеки OpenCV на языке программирования Python. Ниже приведено описание алгоритмов пороговой сегментации Отсу, кластеризации k-means и водораздела (англ. WaterShed), которые использовались в данной работе.

Алгоритм Отсу основан на гистограмме распределения уровней яркости изображения и заключается в поиске порогового значения, разбивающего пиксели на два класса: объектные и фоновые. Вычисление порогового значения основано на итерационном процессе максимизации межклассовой дисперсии:

$$\sigma_b^2 = \omega_1(t)\omega_2(t)[\mu_1(t) - \mu_2(t)]^2 \rightarrow \max,$$

где ω_1 и ω_2 – вероятности первого и второго классов соответственно, μ_1 и μ_2 – средние арифметические значения для каждого из классов, t – пороговое значение, которое принимает целые значения в диапазоне $[1, L]$, где L – максимальный уровень яркости на изображении. Искомое пороговое значение будет соответствовать максимальному

значению межклассовой дисперсии. Принадлежность пикселя к тому или иному классу определяется путем сравнения его яркости с пороговым значением.

Алгоритм водораздела основан на анализе границ и локальных минимумов и использует идею о том, что изображение можно представить в виде карты местности, где резкие изменения яркости будут соответствовать хребтам, а однотонные области - равнинам, иначе говоря, значение длины градиента яркости в каждой точке будет соответствовать значению высоты относительно некоторого уровня. Алгоритм начинается с поиска локальных минимумов. После нахождения минимумов начинается процесс их последовательного заполнения «водой». При объединении двух минимумов между ними ставится условная перегородка, называемая «линией водораздела». В результате работы алгоритма водораздела получают так называемые «водосборные бассейны», которые представляют собой отдельные объекты на изображении, а «линии водоразделов» являются границами данных объектов.

Алгоритм k -средних используется для разделения множества пикселей изображения на заданное количество кластеров и основан на минимизации суммарного квадратичного отклонения точек кластеров от центров кластеров:

$$V = \sum_{i=1}^k \sum_{x \in S_i} (x - \mu_i)^2 \rightarrow \min,$$

где V – суммарное квадратичное отклонение, k – число кластеров, S_i – i -й кластер изображения, i – центр i -го кластера. Перед началом выполнения алгоритма фиксируется количество кластеров и происходит инициализация начальных центров кластеров. Каждый пиксель изображения приписывается к тому кластеру, центр которого наименее удален от данного пикселя. Для вычисления удаленности пикселей чаще всего используется функция евклидова расстояния. После установления принадлежности пикселей к кластерам производится вычисление новых значений центров кластеров путем вычисления среднего арифметического всех точек внутри соответствующего кластера. Условием завершения алгоритма является равенство координат центров кластеров на предыдущей и текущей итерации.

Пример сегментационной маски, на основе алгоритма k -means продемонстрирован на рис. 2а, пример ограничивающей рамки продемонстрирован на рис. 2б.

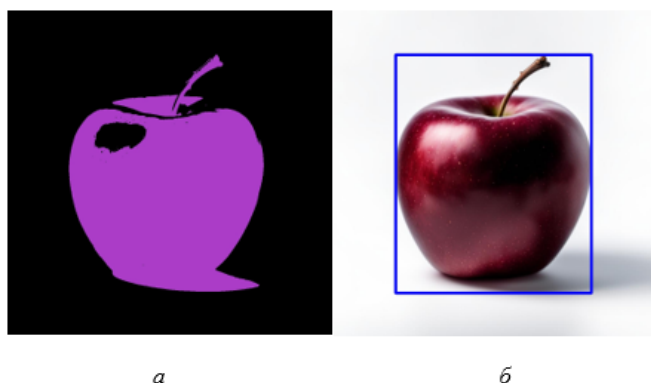


Рис. 2. Сегментационная маска (а), ограничивающая рамка (б).

4. Стилизация изображений

Для того, чтобы улучшить качество сегментации объектов на изображениях определённого стиля в работе был использован алгоритм переноса стиля с помощью нейронных сетей (англ. Neural Style Transfer) [6], реализованный в библиотеке TensorFlow на языке программирования Python. Общий принцип алгоритма был описан Леоном Гатисом [7] и заключается в идее о том, что пиксели исходного изображения

рассматриваются, как настраиваемые параметры в алгоритме градиентного спуска, а качество стилизации изображения рассматривается, как функция, минимум которой будет соответствовать наилучшему результату:

$$J = \alpha J_C + \beta J_S \rightarrow \min.$$

Общий критерий качества разделяют на:

- степень соответствия результата исходному изображению по контенту J_C ;
- степень соответствия результата нужному стилю J_S .

Пример стилизованного изображения продемонстрирован на рис. 3.



Рис. 3. Пример стилизованного изображения.

5. Принцип работы и интерфейс программного обеспечения

Принцип работы программного обеспечения для создания обучающей выборки представлен на рис. 4.

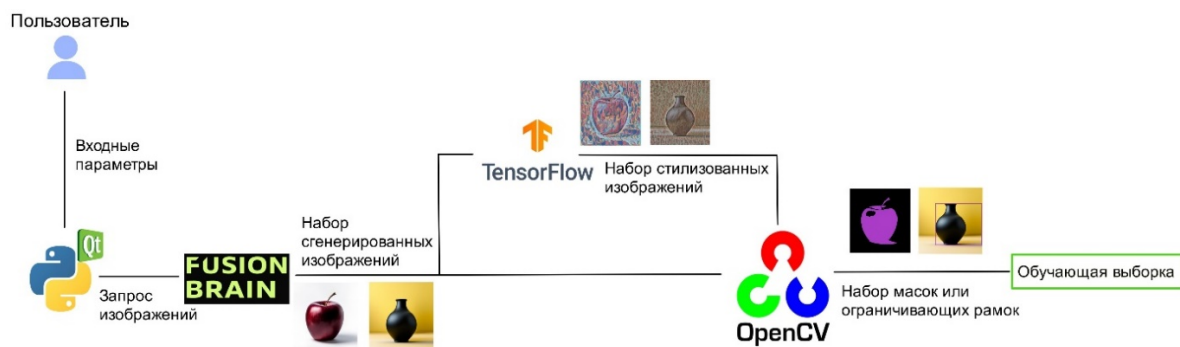


Рис. 4. Принцип работы программного обеспечения.

Интерфейс программного обеспечения представлен на рис. 5.

Введите объект генерации:
яблоко

Введите количество изображений:
20


Начать генерацию изображений

Выберите стиль: style4

Загрузите изображение для стилизации:

Загрузить

Стилизовать изображения



Выберите алгоритм сегментации:
Алгоритм k-средние (k-means)

Выберите способ выделения:
 Pixel Mask
 Bounding box

Запустить алгоритм сегментации

Рис. 5. Интерфейс программного обеспечения.

6. Заключение

В докладе уделено внимание разработке программного обеспечения для создания обучающей выборки, описан способ генерации изображений, алгоритмы сегментации для создания масок и ограничивающих рамок объектов, стилизация изображений.

В дальнейшем планируется усовершенствование качества выделения сегментационной маски объектов с помощью удаления бликов и теней на изображении, а также объединение нескольких классов объектов на одном изображении.

Результаты, полученные в ходе работы, могут быть протестированы при обучении нейронных сетей сегментации, например, DeepLab или детекции объектов, например, YOLO.

Список литературы

1. Стрыгин Д.Д., Лемтюжникова Д.В. Поиск символов на цифровых изображениях голландских натюрмортов // Сборник тезисов 49-й Международной молодежной научной конференции «Гагаринские чтения» (Москва, 2023). М.: Перо, 2023. С. 255-256.
2. Стрыгин Д.Д. Автоматизация анализа символов на цифровом художественном изображении нейросетевым подходом // Труды 16-й Всероссийской мультиконференции по проблемам управления (МКПУ-2023, Волгоград). Волгоград: Волгоградский государственный технический университет, 2023. № 2. С. 272-274.
3. Fusion Brain. API документация. <https://fusionbrain.ai/docs/ru/doc/api-dokumentaciya/> (дата обращения 28.01.2024).

4. Kandinsky 3.0 — новая модель генерации изображений по тексту. <https://habr.com/ru/companies/sberbank/articles/775590/> (дата обращения 28.01.2024).
5. Диффузионные Нейросети — самый актуальный подход к генерации изображений. <https://habr.com/ru/companies/ruvds/articles/689072/> (дата обращения 28.01.2024).
6. Теория стилизации изображений (Neural Style Transfer). https://proproprogs.ru/neural_network/teoriya-stilizacii-izobrazheniy-neural-style-transfer (дата обращения 28.01.2024).
7. Gatys L.A., Ecker A.S., Bethge M. A neural algorithm of artistic style. arXiv preprint arXiv:1508.06576. 2015.