

# “Image Retrieval”

**Цель:** определить расположение конкретного товара на полке магазина.

Международная розничная компания обратилась с запросом на разработку системы при помощи компьютерного зрения (CV) для автоматической проверки выкладки товаров на полках. Поскольку некоторые бренды инвестируют в премиальные места для своей продукции, критически важно обеспечить точность их размещения. В рамках этого проекта вы создадите систему контроля, позволяющую отслеживать работу мерчендайзеров. В случае успешной реализации решение будет масштабировано и внедрено в сеть магазинов компании.

<ПРИМЕР GALLERY ИЗОБРАЖЕНИЯ>



Вы и компания пришли к соглашению, что для демонстрационного проекта достаточно реализовать простую функцию, определяющую расположение элемента. Эта функция должна принимать на вход два изображения:

- Изображение **GALLERY** — аналогичное исходному изображению полки, но без красных рамок вокруг товара.
- Изображение **QUERY** — содержит один товар, вырезанный по контуру.

Например:

<ПРИМЕР QUERY ИЗОБРАЖЕНИЯ>



Результатом работы функции должен быть массив координат (bounding boxes) в формате:  $[(x\_min, y\_min, width, height), \dots]$ , где значения нормализованы в диапазоне  $[0, 1]$  (формат YOLO).

Координаты  $(x\_min, y\_min)$  соответствуют верхнему левому углу ограничивающей рамки и вычисляются по формуле:

$x\_min\_normalized = x\_min\_pixels / \text{ширина изображения}$

$y\_min\_normalized = y\_min\_pixels / \text{высота изображения}$

Если задача окажется слишком сложной, начните с упрощённого варианта — реализуйте классификатор, определяющий наличие элемента **QUERY** на изображении **GALLERY**. В этом случае возвращайте:

- $[(0, 0, 1, 1)]$ , если элемент найден;
- $[]$ , если элемент отсутствует.

Этот подход позволит получить базовый результат и заработать несколько баллов. Следующим шагом станет определение точных координат ограничивающих рамок.

Если задача, наоборот, покажется слишком простой, обратите внимание на реальные сложности, возникающие при работе с фотографиями «производственного качества». Некоторые товары обладают гибкой упаковкой (например, пачки чипсов), что затрудняет использование классических методов компьютерного зрения. В качестве дополнительного challenge-а предлагаем поработать с изображениями йогурта «АКТИВИЯ» (**train\_extreme.jpg** и **template\_extreme.jpg**). Этот этап не обязателен для получения максимальной оценки (100 баллов), но позволит столкнуться с более сложными задачами, выходящими за рамки учебного задания.

#### Данные:

Для того чтобы создать алгоритм, мы предоставляем вам несколько "обучающих примеров". Файл **template\_X\*.jpg** - это запрос (query изображение) к изображению **image\_X.jpg**. Иногда для одного и того же изображения выполняется несколько запросов.

Чтобы протестировать ваш алгоритм, мы будем использовать другой набор изображений, который вам будет недоступен.

#### Требования к отправке:

Чтобы отправить свое решение, вам необходимо:

- (1) Присвоить файлу имя **retrieval.py**. Другие имена файлов распознаваться не будут.
- (2) Присвоить основной конвейерной функции (спецификации ввода-вывода, которые вы можете найти ниже) имя **predict\_image**.
- (3) Загрузите (отправьте) свой файл .py в домашнее задание Image retrieval нашего курса.

#### Спецификации predict\_image:

Функция **predict\_image** принимает два аргумента: gallery и query изображения. Мы загружаем тестовые изображения через

```
img = cv2.imread(path_to_img)
```

и передаем массивы строго в вашу функцию. Выходные данные состоят из одного списка python:

```
boxes_list = predict_image(img, запрос)
```

boxes\_list - это список на языке python, пустой или состоящий из ограничивающих рамок (bounding boxes). Ограничивающий прямоугольник - это кортеж python, состоящий из 4 чисел с плавающей точкой (x\_min, y\_min, width, height). Значения x\_min, y\_min, width и height зависят от формы изображения, т.е. масштабируются в интервале [0, 1] по ширине или высоте изображения.

Также смотрите **\_template\_solution.py** в ZIP-архиве с шаблоном решения.

#### Технические характеристики платформы

Платформа предоставляет вам среду python3 (активированную по умолчанию) и предустановленные пакеты pip, необходимые для решения проблемы: numpy, scipy, scikit-image, scikit-learn, opencv.

#### **Штраф за схожесть кода**

Обратите внимание: мы будем проводить тщательную проверку на схожесть решений. Вы можете делиться своим кодом, однако это сопряжено с риском снижения оценки за домашнее задание.

В случае высокой степени сходства скриптов между участниками, баллы будут снижены **пропорционально количеству совпадающих работ (коэффициент  $N$ , где  $N$  — число участников с одинаковым кодом)**.

### **Штраф за нарушение правил безопасности или утечку информации**

Если вы (сознательно или случайно) обнаружите уязвимости или способы обойти систему, использовать их вы можете **на свой страх и риск**, так как это может привести к снижению баллов за задание.

При выявлении признаков нарушения правил безопасности ваша работа может быть оценена в **0 баллов**.