Машинное зрение — направление в области искусственного интеллекта, которое включает алгоритмы для обнаружения, отслеживания и классификации объектов по их изображениям.

Технология машинного зрения позволяет промышленному оборудованию “видеть” и анализировать объекты, а также взаимодействовать с ними, что в свою очередь помогает повысить качество изготовляемой продукции, сократить расходы на производство, а также упростить задачи.

Компьютер видит не так, как люди. У него нет нашего жизненного опыта и способности так же легко идентифицировать объекты на изображении. Он не способен отличить дом от дерева, не имея каких‑то исходных данных.

*Переход на некст слайд*

На изображении нет ни одного(!) красного пикселя. Просто мы знаем, что клубника красного цвета.

*Переход на некст слайд*

Отсюда вытекают некоторые проблемы:

1. Если не точно известны механизмы и форма передачи данных у человека, как понять, в каком виде передавать информацию компьютеру?
2. А сколько «опыта» должно быть в памяти компьютера, чтобы он делал какие-человеку-хочется выводы? Какой должен быть опыт?

Для компьютера пока существует два глобальных способа определения действий на основе полученных данных: классические методы и машинное обучение.

Классические методы впервые были применены для отличия печатного и рукописного текстов ещё в 1970-х годах. В основе лежит работа с контрастом и цветами изображения, выделение геометрических фигур и границ предметов.

Задача: написать программу, которая на вход получает картинку с геометрическими формами, а после выполнения возвращает ту же картинку с корректными названиями данных фигур.

(Полное решение (вместе с кодом на python и подробным описанием) можно найти [здесь](https://www.pyimagesearch.com/2016/02/08/opencv-shape-detection/).)

— Сначала компьютеру нужен жизненный опыт. В отдельный файл записываются все геометрические формы, которые программа должна уметь распознавать. Также записываются особенности, по которым можно однозначно определить фигуру. Например, у треугольника три стороны, у квадрата четыре, у прямоугольника тоже четыре, но они не равны.

— Дальше код, который загружает картинку в программу, и предобработка: перевод в ч/б, уменьшение размера изображения, размытие и подбор параметров для создания «маски» (все чёрные участки - чёрные, все остальные — белые).

— С помощью готового (взятого из библиотеки) алгоритма на изображении выявляются границы фигур. Информация записывается в виде координат точек, формирующих контуры.

— После сглаживания контуров, можно достать количество и длину сторон фигуры.

— Наконец, по особенностям каждой формы (количество, длина сторон) программа может определить её название.

*Переход на некст слайд*

Ещё один подход к решению задач — машинное обучение. Это иной способ накопления опыта: набор методов, которые умеют замечать и запоминать взаимосвязи между признаками.

В основе машинного обучения лежат три одинаково важных компонента:

* **Данные.**Собираются всевозможными способами. Чем больше данных, тем эффективней машинное обучение и точнее будущий результат.
* **Признаки.**Определяют, на каких параметрах строится машинное обучение.
* **Алгоритм.**Выбор метода машинного обучения (при условии наличия хороших данных) будет влиять на точность, скорость работы и размер готовой модели.

Для простоты восприятия типы машинного обучения принято разделять на три категории:

* обучение с учителем (supervised learning);
* Этот тип максимально похож на процесс познания окружающего мира ребенком, только в роли малыша выступает алгоритм. Данные, подготовленные для анализа, изначально содержат правильный ответ, поэтому цель алгоритма — не ответить, а понять, «Почему именно так?» путем выявления взаимосвязей. Результатом становится способность выстраивать корректные прогнозы и модели.
* обучение без учителя (unsupervised learning);
* Для данного типа обучения ключевым понятием является паттерн — обрабатывая значительные массивы данных, алгоритм должен сперва самостоятельно выявлять закономерности. На следующем этапе на основе выявленных закономерностей машина интерпретирует и систематизирует данные.
* обучение с подкреплением (reinforcement learning).
* Принципы обучения с подкреплением заимствованы из психологических экспериментов: машина пытается найти оптимальные действия, которые будет предпринимать, находясь в наборе различных сценариев. Эти действия могут иметь как краткосрочные, так и долгосрочные последствия, а от алгоритма требуется обнаружить эти связи.

OpenCvSharp

*Новый слайд*

OpenCvSharp - это обёртка над OpenCV для использования библиотеки в C# проектах.

Так как это просто обёртка поговорим о OpenCv. **OpenCV** — библиотека алгоритмов компьютерного зрения, обработки изображений и численных алгоритмов общего назначения с открытым кодом на С++

*Новый слайд*

Сейчас структура OpenCV — это множественные [модули](https://docs.opencv.org/4.5.4/modules.html) для разных целей:

* хранения математических функций и вычислений, алгебры и структур данных;
* хранения моделей для машинного обучения;
* ввода и вывода картинок или видео, чтения и записи в файл;
* обработки изображения;
* распознавания примитивов;
* детектирования объектов — лиц, предметов и других;
* отслеживания и анализа движений на видео;
* обработки трехмерной информации;
* ускорения работы библиотеки;
* хранения устаревшего или еще не готового кода и других.

Преимущества:

### Активное сообщество

### Бесплатный доступ

### Обилие алгоритмов

OpenCV включает более 2500 инструментов и алгоритмов компьютерного зрения и машинного обучения

### Высокая скорость

### Возможность работы в реальном времени

### Недостатки: Сложность в освоении

Чтобы хорошо понимать все возможности OpenCV, нужно знать теории компьютерного зрения и машинного обучения.

### Отсутствие кодов обработки ошибок

Если возникла ошибка, в OpenCV бывает сложно понять, где именно.

### Ориентированность на большие платформы

OpenCV работает на масштабных платформах. Если запустить ее на микроконтроллере, одноплатном компьютере, производительность будет невысокой.