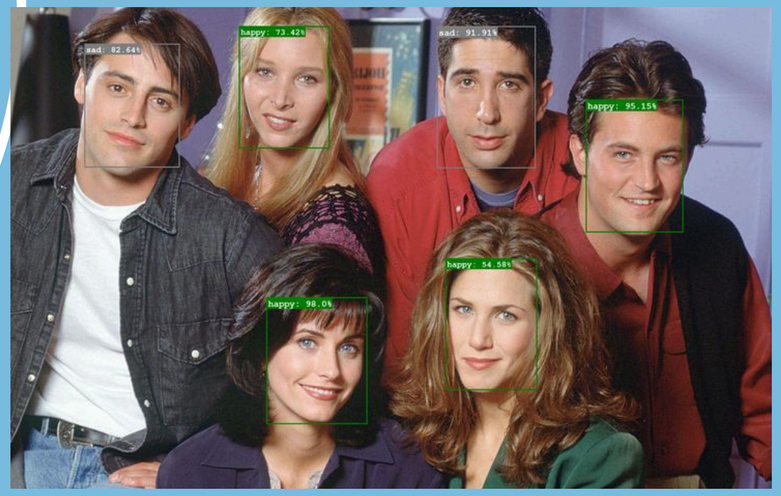
Ministry of Science and Higher Education of the Republic of Kazakhstan
L.N. Gumilyov Eurasian National University

Faculty of Information Technology
Department of Information Systems

ALGORITHMS FOR DETECTION

Done by: Zhumabek A.R

INTRODUCTION



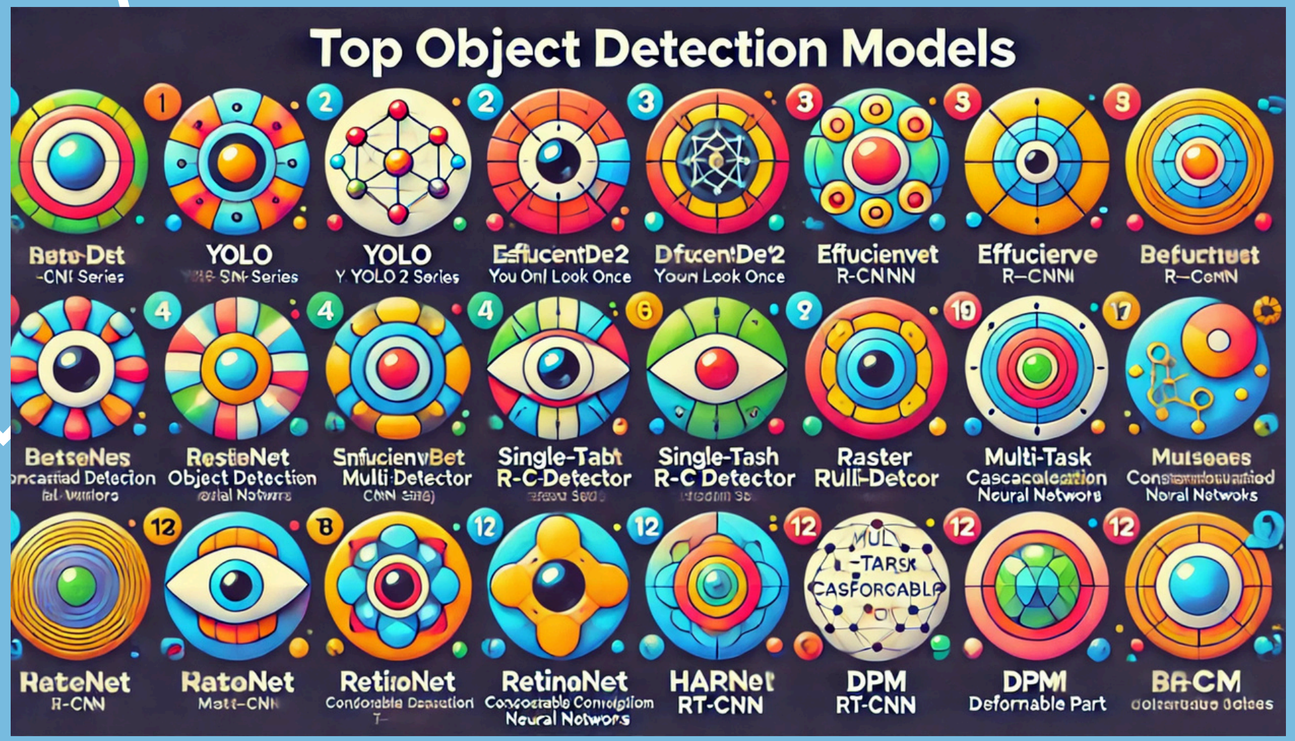
Object detection is a key component of many deep learning models and has undergone a number of revolutionary transformations in recent years. Object detection refers to the process of identifying objects in an image and extracting a bounding box for each object. Object detection algorithms are today used in many fields, for example autonomous driving, security cameras, robotics and in almost any vision-related applications and emerging trends such as Amazon Go grocery store without a cashier.

There are a number of deep learning algorithms that address the problem of object detection. These object detectors consist of three main components:

- **Backbone** for extracting features from the image
- **Feature network** that takes features from the backbone and outputs features to represent the characteristics of the image
- **Final** (class/box) network that uses these features to predict the class and location of an object

VIOLA-JONES VS SSD VS CASCADE R-CNN

In computer vision, there are many other algorithms for object detection and I chose three of them: (to compare the cascade and single-stage approaches)



01 Viola-Jones

02 SSD

03 Cascade R-CNN

VIOLA JONES

Viola-Jones algorithm is one of the most popular methods for detecting faces in images due to its high speed and efficiency. It was developed by Paul Viola and Michael Jones in 2001 and is still used for real-time object detection. The algorithm has a low probability of false positives and works well even at small face angles (up to 30°).

Basic operating principles:

- **Haar features** - simple patterns that compare brightness in rectangular regions of an image, which helps to detect eyes, nose, mouth and facial contours.

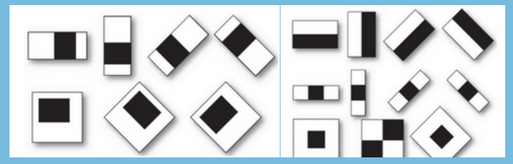


Fig. 1. Haar feature primitives and Additional Haar features

- **Integral image** - a special representation that speeds up the calculation of the sum of pixels in any rectangular region of an image.

$$I(x,y) = \sum_{x' \le x, y' \le y} i(x', y'),$$

- **AdaBoost** - a method for strengthening weak classifiers by selecting the most significant Haar features and combining them into one strong classifier.
- **Cascade** of classifiers - a multi-level structure where irrelevant regions are quickly pruned out in the initial stages and more accurate analysis is performed in later stages.

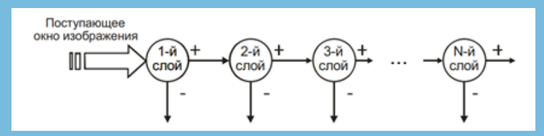
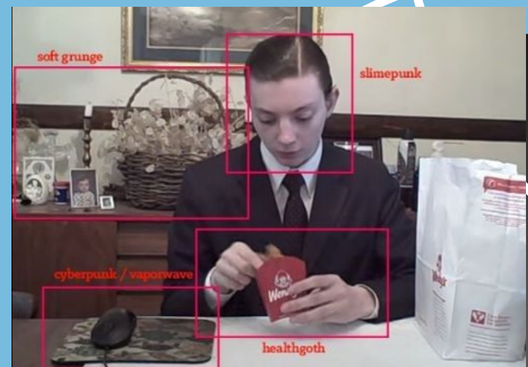


Fig. 2. Structure of the cascade detector

This combination allows Viola-Jones to perform in real time, remaining one of the fastest face detection algorithms available



SSD (SINGLE SHOT MULTIBOX DETECTOR)

SSD (Single Shot MultiBox Detector) is a popular object detection algorithm designed for high speed and accuracy. Unlike Viola-Jones, which uses a cascade approach, SSD performs object detection in a single step, making it faster and more efficient on modern devices.

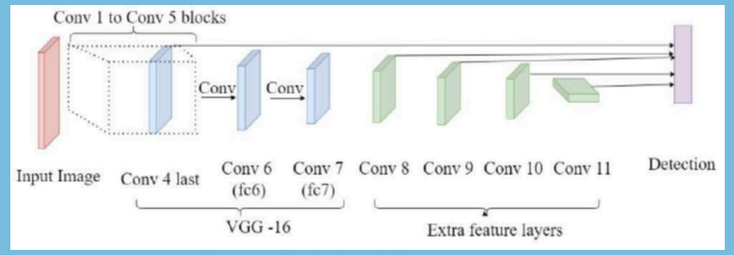


Fig 3. The architecture of SSD multi-box detector

Figure 3 shows how an SSD can be constructed using VGG-16 and a few additional layers. First VGG-16 processes the image, then additional convolution layers help to find objects of different sizes. The algorithm uses a fixed number of bounding boxes with different aspect ratios. Unlike R-CNN, it makes predictions in a single pass, making it fast. There are two versions: the SSD300, which is faster but less accurate, and the SSD512, which is more accurate but slower. The SSD is well suited for real-time tasks because it balances speed and accuracy.

Key features:

- Single-stage detection: SSD predicts object frames and their classes simultaneously using a convolutional neural network (CNN). This is different from the Viola-Jones cascading approach, which goes through several classification stages.
- Multi-level feature maps: SSD uses feature maps of different sizes, which allows detection of both small and large objects in an image.
- Anchor Boxes (Anchor Boxes): For each level of the feature map, frames of different shapes and sizes are predicted, which improves detection accuracy.
- High Speed: Due to its single-stage architecture, SSD can operate in real-time, making it popular in mobile and surveillance applications.

VIOLA-JONES COMPARISON WITH SSD

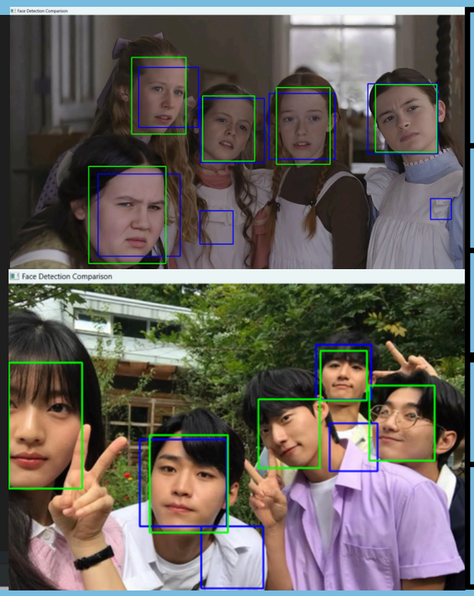
Speed
Both algorithms operate in real time, but SSD achieves this through a deep neural network, while Viola-Jones achieves it through a cascading structure and simple features.

Flexibility
SSD handles changes in angle, scale and lighting better, while Viola-Jones is more sensitive to such changes.

Accuracy
SSD shows higher accuracy due to the use of CNNs, while Viola-Jones is limited by the use of Haar features.

```python
import cv2
import time
# === Загрузка моделей ===
# Viola-Jones (Haar Cascade)
face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade_frontalface_default.xml')
# SSD (Caffe модель для обнаружения лиц)
net = cv2.dnn.readNetFromCaffe(
    "C:/Users/User/PycharmProjects/pythonProject2/Vio/deploy.prototxt",
    "C:/Users/User/PycharmProjects/pythonProject2/Vio/res10_300x300_ssd_iter_140000.caffemodel"
)
# === Функция для тестирования Viola-Jones ===
def test_viola_jones(image):  1 usage
    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
    start = time.time()
    faces = face_cascade.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=5)
    end = time.time()
    return faces, end - start
# === Функция для тестирования SSD ===
def test_ssd(image):  1 usage
    h, w = image.shape[:2]
    blob = cv2.dnn.blobFromImage(image,  scalefactor: 1.0,  size: (300, 300),  mean: (104.0, 177.0, 123.0))
    net.setInput(blob)
    start = time.time()
    detections = net.forward()
    end = time.time()
    faces = []
    for i in range(detections.shape[2]):
        confidence = detections[0, 0, i, 2]
        if confidence > 0.5:  # Фильтруем по порогу уверенности
            box = detections[0, 0, i, 3:7] * [w, h, w, h]
            faces.append(box.astype("int"))
    return faces, end - start
```

ject2 › Vio › VioSSD.py



Algotithm	faces	time
Viola-Jones	7	0.3057 sec
SSD	5	0.0449 sec
Viola-Jones	4	0.2383 sec
SSD	5	0.0415 cek

SSD is faster and better able to adapt to changing conditions such as angle, illumination and scale because it uses a deep neural network, whereas Viola-Jones is slower due to cascading classifiers and more sensitive to external changes. Practical results confirm this: SSD was significantly faster and more accurate in identifying faces in the first image, whereas Viola-Jones missed one. However, Viola-Jones found more faces in the second case, but this could be due to false positives. Overall, SSD is faster but can miss faces, while Viola-Jones is slower and sometimes mistakenly detects extra objects.

CASCADE R-CNN

Cascade R-CNN is an advanced object detection algorithm designed to improve accuracy by utilizing a multi-level cascade of detectors. Unlike Viola-Jones, which uses a cascade of classifiers with fixed thresholds, Cascade R-CNN uses a sequence of multiple detectors with increasing severity.

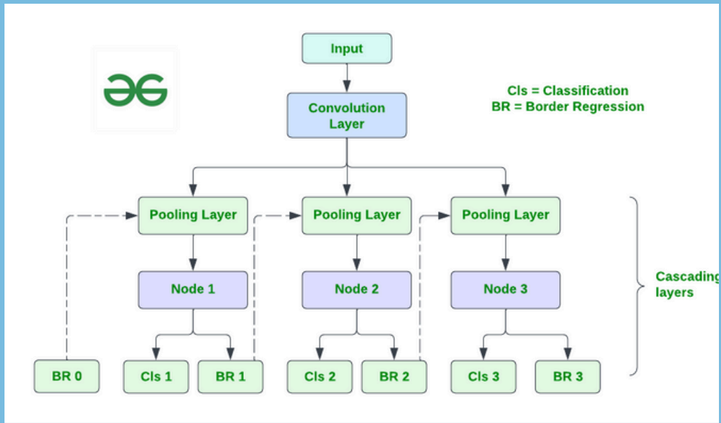


Fig 4. The architecture of Cascade R-CNN

Architecture of Cascade R-CNN

Cascade R-CNN proposed as a multi-stage object detection framework designed to enhance the quality of object detectors by addressing challenges such as noisy detections and performance degradation with increasing Intersection over Union (IoU) thresholds. The architecture of Cascade R-CNN involves a sequence of detectors trained with progressively higher IoU thresholds, making them more selective against close false positives.

Main features:

- Multi-level cascade: The algorithm consists of several stages, where each subsequent detector processes more complex cases using the results of the previous one. This improves the quality of the predicted bounding boxes.
- Progressive Learning: Each detector is trained on more stringent data, which reduces false positives and improves accuracy.
- Faster R-CNN-based architecture: Cascade R-CNN is based on Faster R-CNN, but with additional cascading stages to improve object localization accuracy.
- Flexible and scalable: Cascade R-CNN can easily adapt to different levels of object and background complexity due to its multi-level structure.

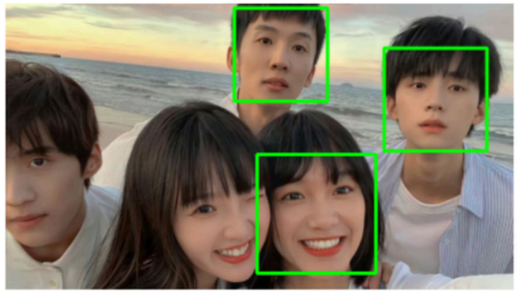
VIOLA-JONES COMPARISON WITH CASCADE R-CNN

Cascade Approach Both algorithms use cascade structures, but Cascade R-CNN applies multi-level rigor to deep neural networks whereas Viola-Jones applies it to simple Haar classifiers.	**Accuracy** Cascade R-CNN is significantly more accurate due to the use of CNNs and progressive learning, whereas Viola-Jones is limited to simple features.
Speed Viola-Jones is faster on simple tasks (e.g., face detection on a webcam), while Cascade R-CNN is more resource intensive but more accurate in complex scenes.	**Flexibility** Cascade R-CNN handles different scales, angles, and object lighting better. Viola-Jones is more sensitive to these changes.

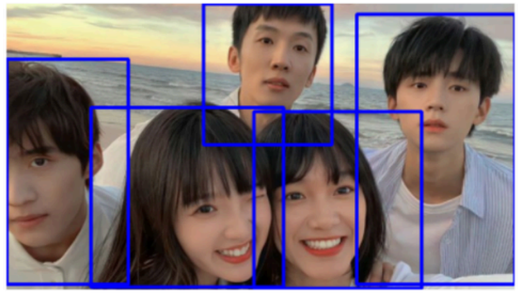
```python
import cv2
import numpy as np
import time
import torch
import torchvision.transforms as T
import matplotlib.pyplot as plt
from torchvision.models.detection import fasterrcnn_resnet50_fpn

from google.colab import drive
drive.mount('/content/drive')

image_path = "/4.jpg"

model = fasterrcnn_resnet50_fpn(pretrained=True)
model.eval()

def detect_faces_viola_jones(image_path):
    face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + "haarcascade_frontalface_default.xml")

    image = cv2.imread(image_path)
    if image is None:
        print("Mistake")
        return 0, 0, None

    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

    start_time = time.time()
    faces = face_cascade.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=5, minSize=(30, 30))
```

```
Viola-Jones  3 faces, Time: 0.1512 sec
Cascade R-CNN  5 faces, Time: 9.7436 sec
```

Viola-Jones Detection

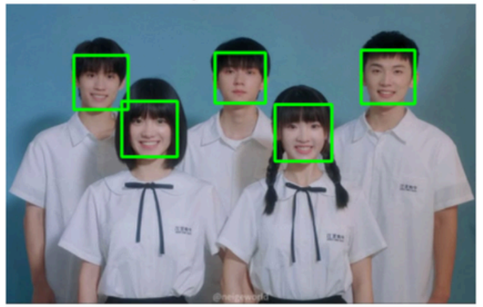


Cascade R-CNN Detection

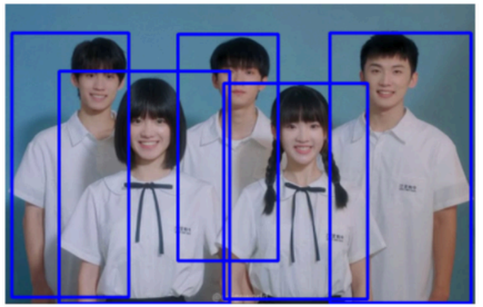


```
Viola-Jones  5 faces, Time: 0.1607 sec
Cascade R-CNN  5 faces, Time: 9.8098 sec
```

Viola-Jones Detection



Cascade R-CNN Detection



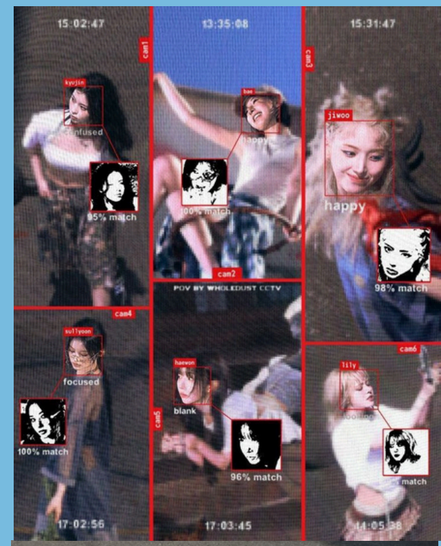
Practical results confirmed the theoretical assumptions. Viola-Jones is much faster because it uses cascading Haar classifiers, whereas Cascade R-CNN requires more time due to its deep neural network architecture. In the tests, Viola-Jones handled in 0.1512-0.2845 seconds while Cascade R-CNN took 6.0364-9.7436 seconds, which proves that Viola-Jones is suitable for fast processing but is less accurate. In terms of the number of faces found, Viola-Jones found 2-3 faces, while Cascade R-CNN found 5-7 faces, which confirms its high accuracy. Practice proved the theory: Cascade R-CNN is more accurate but slower, while Viola-Jones is faster but can miss faces. If speed is important, it is better to use Viola-Jones, and if you need accuracy, use Cascade R-CNN.

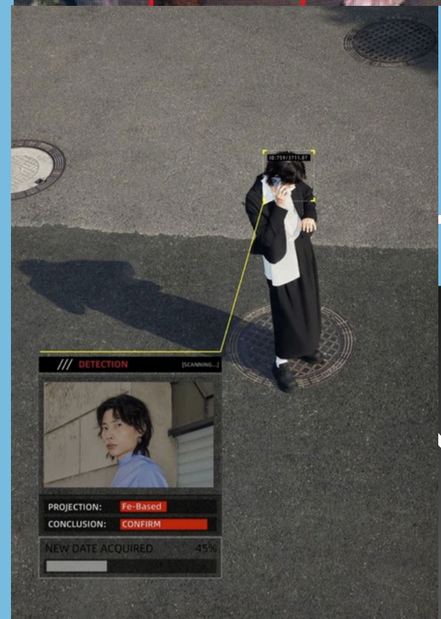
COMPARISON TABLE: VIOLA-JONES VS SSD VS CASCADE R-CNN

Criterion	Viola-Jones	SSD (Single Shot Detector)	Cascade R-CNN
Method	Haar Cascade Classifiers	Convolutional Neural Network (CNN)	Deep Convolutional Neural Network (CNN)
Processing Speed	Very fast (0.1–0.3 sec)	Medium (0.5–2 sec)	Slow (6–10 sec)
Accuracy (Faces Detected)	Low (2–3 faces)	Medium (4–6 faces)	High (5–7 faces)
Robustness to Lighting and Rotation	Poor	Moderate	High
False Positives Sensitivity	High	Moderate	Low
Suitable for Real-Time Applications	Yes	Yes, but GPU recommended	No, too slow
Computational Demand	Low (Runs on CPU)	Medium (GPU Recommended)	High (Requires powerful GPU)
Primary Use Case	Fast but simple face detection	Balance between speed and accuracy	High-precision detection but slow

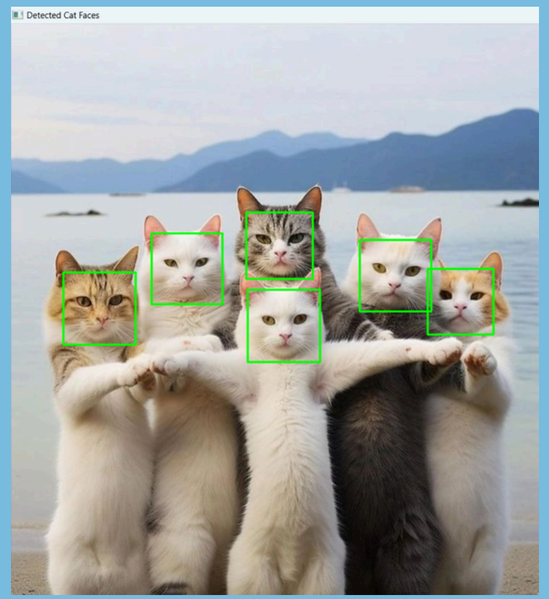
CONCLUSION

Viola-Jones - fast but less accurate, suitable for simple tasks. SSD - good balance between speed and accuracy, can be used in real time. Cascade R-CNN - the most accurate, but very slow, requires powerful calculations.
And as usual, the choice of algorithm depends on the task: if we need speed - Viola-Jones, if we need balance - SSD, if we need maximum accuracy - Cascade R-CNN.





THANK YOU



References

- www.reallygreatsite.com
- https://www.researchgate.net/publication/365019554_Various_object_detection_algorithms_and_their_comparison
- https://www.geeksforgeeks.org/how-single-shot-detector-ssd-works/
- https://medium.com/0xcode/the-viola-jones-face-detection-algorithm-3eb09055cfc2
- https://arxiv.org/pdf/1906.09756v1