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

MEDIAPIPE

Done by: Zhumabek A.R
Check: Zhukabaeva T.K

INTRODUCTION

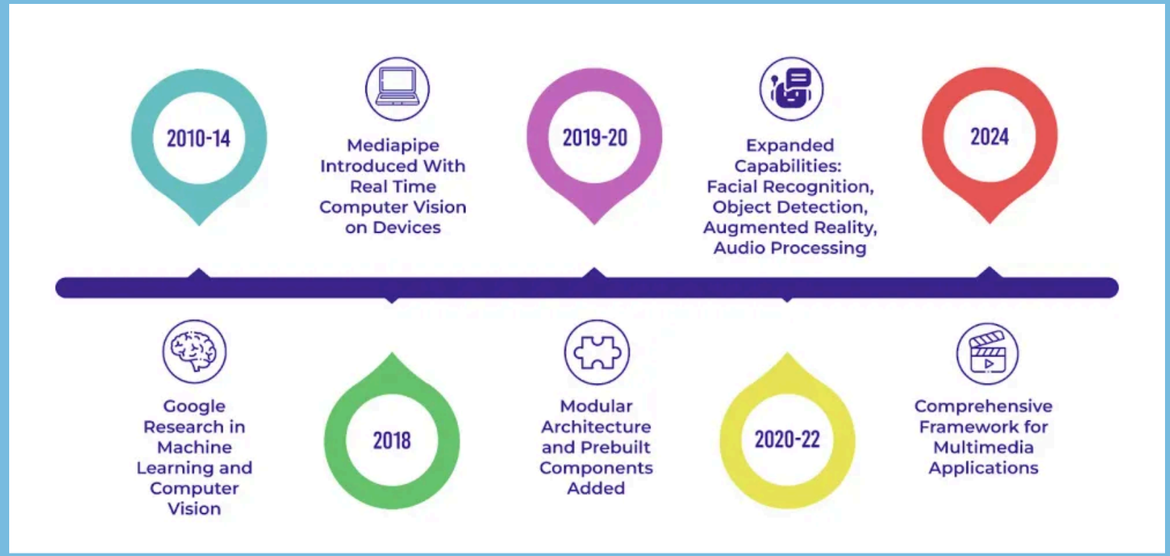
In today's world of computer vision and multimedia processing, it is critical to have tools that allow you to efficiently build and deploy machine learning models in real time. MediaPipe is a powerful open-source framework from Google that provides out-of-the-box solutions for building such ML pipelines.

Thanks to its modular architecture and graph-based approach, MediaPipe enables the development of complex data processing pipelines consisting of individual components - “Calculators”. They perform sequential processing of information such as video, audio and text analysis, making the system flexible and adaptive.

The platform supports running on web applications, mobile devices (iOS, Android) and even embedded systems, making it a versatile tool for developing computer vision applications. In this article, we will take a closer look at how MediaPipe works and its key features.

MEDIAPIPE OVER THE YEARS

2010-2014: Google's research in machine learning and computer vision. In 2012, MediaPipe was used to analyze real-time video and audio on YouTube.
2018: Introduced a modular architecture with pre-built components (Calculators), which simplified the creation of ML Pipelines. MediaPipe began solving the challenges of deploying complex computer vision models on smartphones and low-power devices.
2019-2020: Official release of MediaPipe with the ability to process live video directly on devices.
2020-2022: Expanded capabilities - adding facial recognition, object detection, augmented reality and audio processing.
2024: MediaPipe becomes a complete framework for multimedia applications, offering powerful and flexible tools for developers.
This framework continues to evolve, making machine learning available across platforms.



CORE FEATURES AND TECHNOLOGIES

🔹 Hardware acceleration:

One of the key features of MediaPipe is the use of graphics processing units (GPUs) for accelerated processing. This is especially useful for complex tasks that require high processing power, such as real-time video processing, multithreading video streams, or working with multiple computer vision models simultaneously.

🔹 OpenCV Integration:

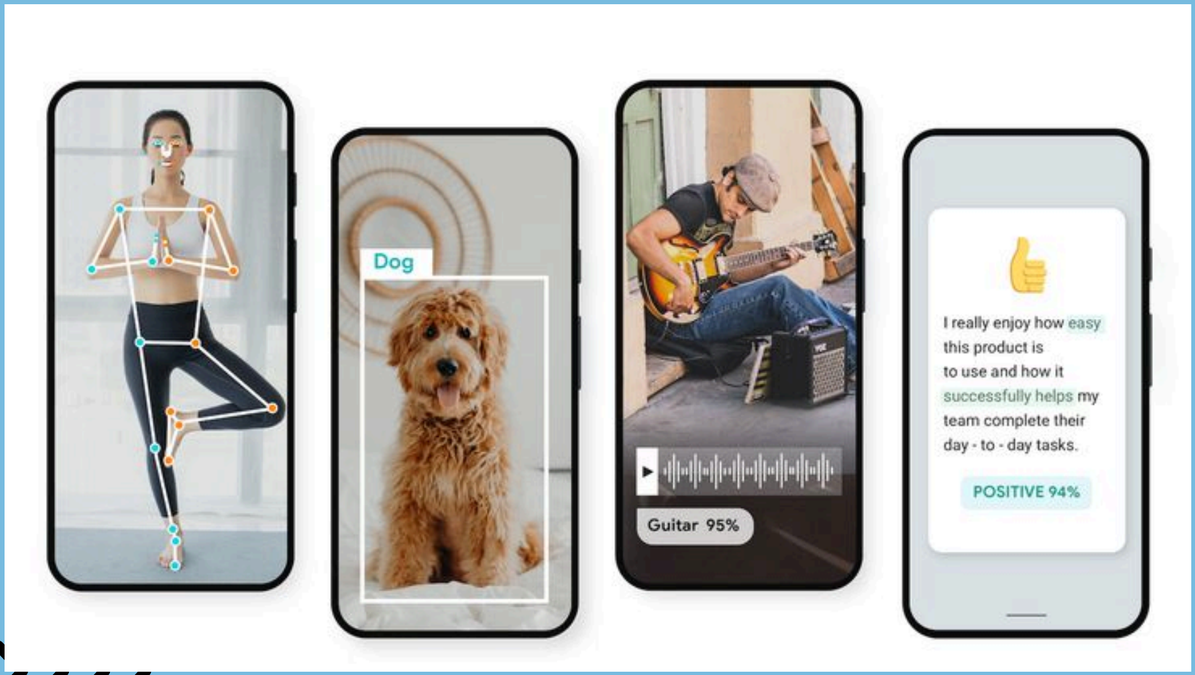
MediaPipe utilizes OpenCV, a powerful computer vision library with tools for image and video processing. This makes it easy to implement video capture, preprocessing and rendering.

🔹 Compatibility with TensorFlow:

MediaPipe integrates tightly with TensorFlow, making it easy to use pre-trained and custom machine learning models. This makes the tasks of recognizing faces, gestures, objects, and even speech processing much easier.

🔹 Multi-language support:

MediaPipe can be used with C++, Java, and Python, making it easy to integrate into different projects.



Other important features of MediaPipe

✅ Ready-made pre-trained models - Simplifies the integration of ML solutions into applications.

✅ MediaPipe Model Maker - Allows you to customize models for specific data.

✅ Model Visualization and Evaluation - Built-in tools for testing solutions right in your browser.

✅ Optimized for on-device performance - MediaPipe is designed to run efficiently without a cloud connection, ensuring low latency and high performance on smartphones and embedded systems.

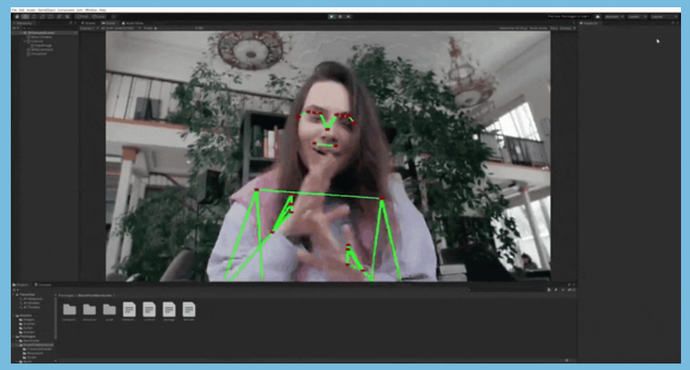
MEDIAPIPE USE CASES

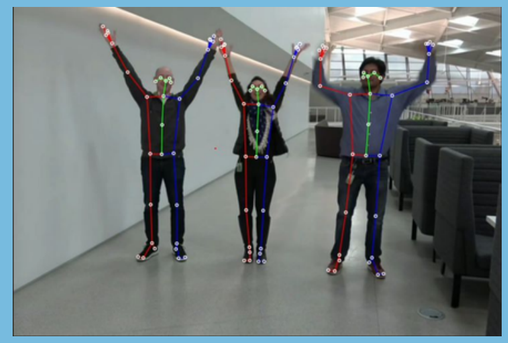
MediaPipe opens up many opportunities in different areas due to its powerful features. Let's take a look at a few key examples of its use.

1️⃣ Human pose detection 🏋️

MediaPipe is used extensively in fitness, sports and medicine due to its accurate human posture detection. This tool tracks joint and body movements in real time, making it useful for:

✅ Developing virtual fitness applications with personalized workouts.
✅ Analyzing sports movements to improve technique.
✅ Assisting in rehabilitation and physical therapy by providing feedback on proper movement patterns.





2️⃣ Improving video calling 🎥

In the era of remote working and online communication, MediaPipe improves video calls with:

✅ Dynamic Frame Adjustment - the camera automatically keeps the person in the center of the screen even if they move.
✅ Gesture control - users can switch slides, adjust volume, or perform other actions using gestures, making interaction more natural.

3️⃣ Creating AR Filters 🤳

MediaPipe is heavily used in the development of augmented reality (AR) filters popular on Instagram, Snapchat, and TikTok. The process includes:

✅ Facial recognition for accurate feature detection.
✅ Adding effects such as virtual masks, makeup, or animated overlays that respond to facial expressions.
✅ Interactive filters for marketing, entertainment, and creative projects.

THE STRUCTURE OF MEDIAPIPE'S WORK

MediaPipe works as a streaming data processing framework, where information passes through a sequence of calculators (Calculators). The whole processing is represented as a graph consisting of nodes (calculators) and data flows between them.

Main elements of MediaPipe structure

1️⃣ Calculators 🛠️

- These are the basic computational units that perform specific tasks (e.g. face detection, image processing).
- Each calculator takes input data, processes it, and passes it on.
- They can use GPUs and TensorFlow to speed up computation.

2️⃣ Data Streams (Streams) 🔄

- Link calculators to each other.
- They are a sequence of data packets (for example, frames of video).
- Provide asynchronous processing, which makes MediaPipe fast.

3️⃣ Processing Graph (Graph) 📊

- This is a structure that links calculators and data streams.
- It allows flexible customization of processing: you can add new modules or change the order of calculations.
- For example, the graph can include face recognition → AR filter overlay → rendering.

🔹 How does MediaPipe work in practice?

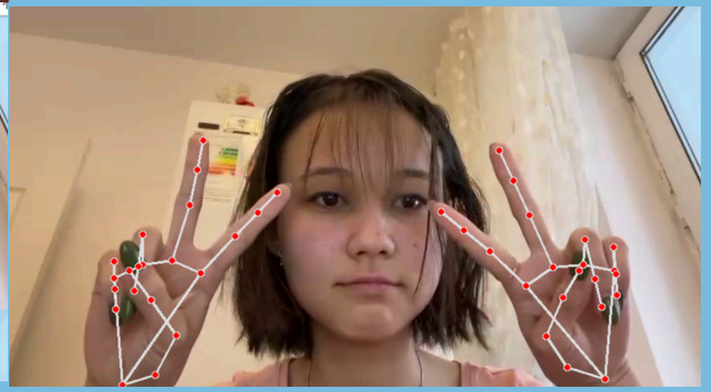
📥 Receiving input data (image, video, audio).

🔄 Transferring data via streams to calculators.

💻 Processing on CPU or GPU (analyzing images, applying filters, etc.).

📤 Outputting the result (visualization, sending data to an application).

```python
import cv2
import mediapipe as mp
from google.colab.patches import cv2_imshow

mp_hands = mp.solutions.hands
mp_drawing = mp.solutions.drawing_utils

video_path = "/v1.mp4"
cap = cv2.VideoCapture(video_path)


with mp_hands.Hands(min_detection_confidence=0.5, min_tracking_confidence=0.5) as hands:
    while cap.isOpened():
        ret, frame = cap.read()
        if not ret:
            break


        frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
        results = hands.process(frame_rgb)


        if results.multi_hand_landmarks:
            for hand_landmarks in results.multi_hand_landmarks:
                mp_drawing.draw_landmarks(frame, hand_landmarks, mp_hands.HAND_CONNECTIONS)


        cv2_imshow(frame)

cap.release()
cv2.destroyAllWindows()
```



This code uses MediaPipe Hands and OpenCV to detect and track hands in a video.
Import necessary libraries:

- cv2 - OpenCV library for working with images and video.
- mediapipe - MediaPipe library that provides ready-made machine vision models.

Initializing MediaPipe Hands

- mp_hands - load the hand detection model from MediaPipe.
- mp_drawing - drawing utility for drawing key points and finger joints.

Opening a video

- video_path is the path to the video file.
- cv2.VideoCapture(video_path) - open the video file for processing.

Processing each frame

Hand detection

- translate the image from BGR to RGB (MediaPipe works with RGB).
- hands.process(frame_rgb) - pass the frame to the neural network that searches for hands.

Drawing key points on hands

- draw finger joints and connections in the image.

Video display

- Displays a processed frame with hand keypoints drawn.

What does this code do?
Opens the video.
Detects only hands using MediaPipe Hands.
Draws key points of the fingers.
Shows the video frame by frame in Google Colab.

CONCLUSION

MediaPipe is a powerful and user-friendly computer vision library that makes it easy to implement real-time facial, pose, hand, and other AI features. Thanks to its cross-platform and high performance, MediaPipe has applications ranging from fitness applications and AR filters to gesture control and video analytics. Its ease of integration makes it a great tool for both novice and experienced developers, opening up a wide range of opportunities to create innovative solutions.

THANK YOU

References

1. https://habr.com/ru/companies/oleg-bunin/articles/735024/
2. https://blog.roboflow.com/what-is-mediapipe/#mediapipe-use-cases
3. https://medium.com/swlh/a-review-of-googles-new-mobile-friendly-ai-framework-mediapipe-25d62cd482a1