BSDS-208 : Image and Video Analytics

Course Instructor - Mr. Aishwary Shukla

Gauri Sharan - BSc Data Science, Semester 4

June 11, 2024

Table of contents

- MediaPipe is a cross-platform framework developed by Google for building multimodal machine learning pipelines. It supports a variety of domains such as vision, audio, and text.
- The framework is designed to facilitate the deployment of machine learning models on various devices including mobile phones, desktops, and the web, enabling real-time processing capabilities.
- MediaPipe offers a comprehensive set of pre-built solutions and tools to streamline the development process, making it easier for developers to implement advanced ML models in their applications.

History

- MediaPipe originated as an internal tool at Google, aimed at streamlining the development and deployment of machine learning models across various platforms.
- Recognizing its potential, Google released MediaPipe as an open-source project to enable the broader research and developer community to benefit from its capabilities.
- Since its release, MediaPipe has seen continuous development and integration of new features, expanding its applications in areas like hand tracking, face detection, and gesture recognition.

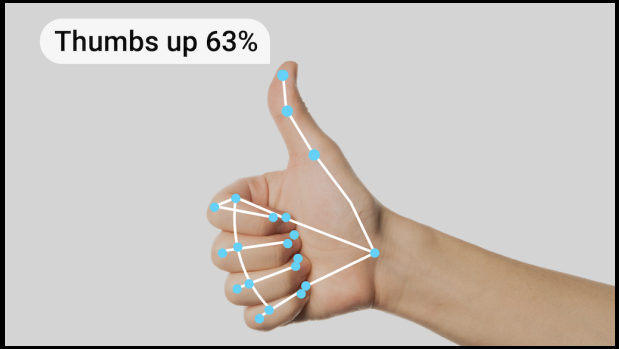

Figure: Hand gesture recognition using MediaPipe's Models

Research Analysis

- MediaPipe's gesture recognition models are grounded in cutting-edge research in computer vision and machine learning, utilizing convolutional neural networks (CNNs) and other advanced techniques.
- The models leverage extensive datasets to learn the nuances of hand movements and gestures, ensuring high accuracy and robustness.
- Research has shown that MediaPipe's models can achieve real-time performance with minimal latency, making them suitable for interactive applications.

Insights

- MediaPipe's models exhibit high accuracy in detecting and classifying hand gestures, which is crucial for applications requiring precise and reliable gesture recognition.
- The framework's cross-platform compatibility ensures that applications can be deployed on a wide range of devices without significant modifications.
- MediaPipe offers customization tools, allowing developers to fine-tune models according to specific datasets and use cases, enhancing the model's performance in targeted applications.

Components and Descriptions of Model

- **HandGestureClassifier**: This model classifies gestures based on the detected landmarks of the hand. It maps the positions of key points on the hand to specific gestures using trained classifiers.
- **GestureRecognizer**: This model combines hand landmark detection with gesture classification. It first identifies key points on the hand and then uses these points to recognize specific gestures, providing a comprehensive solution for gesture recognition.
- Both models are built upon MediaPipe's robust hand tracking solution, which accurately detects and tracks hand landmarks in real-time.

Components and Descriptions of Model

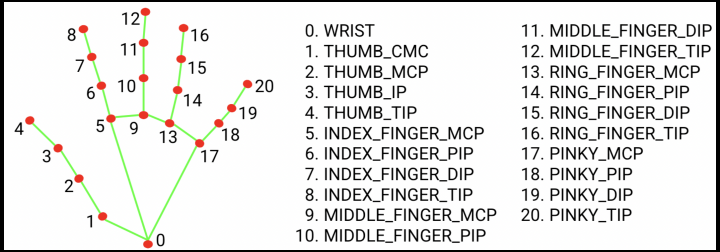

Figure: HandGestureClassifier Model Landmarker Bundle

Architecture and Working of Model

- The architecture of MediaPipe's gesture recognition models consists of several stages, including hand detection, landmark localization, and gesture classification.
- Hand detection is performed using a CNN that identifies the presence of hands in the input image or video stream.
- Once hands are detected, a separate model locates the precise positions of key landmarks on the hands.
- These landmarks are then fed into a gesture classifier, which uses trained algorithms to recognize and classify the gestures.
- This multi-stage pipeline ensures high accuracy and real-time performance, making the models suitable for interactive applications.

Training, Testing, Validation Hands-on

- Training of the models involves using large datasets of hand images annotated with key landmarks and corresponding gestures. The models learn to identify patterns and features associated with different gestures.
- Testing is conducted on separate datasets that were not used during training to evaluate the model's performance and generalization capabilities.
- Validation involves cross-validation techniques where the data is split into multiple subsets. The model is trained on some subsets and validated on others to fine-tune parameters and improve performance.
- This rigorous process ensures that the models perform well in real-world scenarios and can handle diverse inputs effectively.

Real Life Applications

- **Sign Language Recognition**: MediaPipe's models can be used to develop applications that interpret sign language, facilitating communication for the hearing impaired.
- **Human-Computer Interaction**: Gesture recognition enables more intuitive and natural interactions with computers and smart devices.
- **Virtual Reality and Augmented Reality**: In VR and AR applications, gesture recognition allows users to interact with virtual environments in a more immersive and interactive way.
- **Interactive Gaming**: Gesture-based controls enhance gaming experiences, making them more engaging and interactive.

Comparison with Other Models

- MediaPipe models offer superior real-time performance compared to traditional gesture recognition models, which may struggle with latency issues.
- The framework's cross-platform support provides an advantage over models designed for specific environments, such as desktop-only or mobile-only solutions.
- MediaPipe's customization options allow developers to adapt the models to their specific needs, providing more flexibility and efficiency compared to other less adaptable models.

Customization

- MediaPipe models can be customized using the MediaPipe Model Maker, which simplifies the process of training models with custom datasets.
- Developers can retrain models on their specific datasets to improve accuracy and performance for their particular use cases.
- Customization involves minimal code changes, making it accessible even to those with limited machine learning expertise.
- This flexibility allows developers to create highly specialized models tailored to their specific needs.

Python Code Example


```python
import mediapipe as mp
from mediapipe.tasks.python import vision

# Load the hand gesture recognizer
gesture_recognizer = vision.GestureRecognizer.create_from_model_path('path_to_model.tflite')

# Use the recognizer with an image
image = mp.Image.create_from_file('hand_image.jpg')
recognition_result = gesture_recognizer.recognize(image)

# Print recognized gestures
for gesture in recognition_result.gestures:
    print(f'Gesture: {gesture.name}, Confidence: {gesture.score}')
```

Project

Project Link: **github.com/gaurisharan/gesture-recognition-mediapipe**
Observed Output:

Figure: Gesture Recognition Model Output

References

- MediaPipe Framework. *Official Website*. https://ai.google.dev/edge/mediapipe/solutions/vision/gesture_recognizer
- MediaPipe Solutions. *Official website*. https://ai.google.dev/edge/mediapipe/solutions/customization/gesture_recognizer

Thank You

Hope you liked this presentation.

Gauri Sharan
Student, School of Data Science
AAFT Noida (Shobhit University)
BSc Data Science 2022-25
Semester 4, 2024

- LinkedIn: **linkedin.com/in/gauri-sharan**
- GitHub: **github.com/gaurisharan**
- Mail: **gaurisharan123@gmail.com**