

Filter

- ▶ Gesture recognition
- ▶ Hand landmark detection
- ▶ Image embedding
- ▶ Face detection
- ▶ Face landmark detection
- ▼ Pose landmark detection
 - Overview
 - Android
 - Web
 - Python
 - iOS
- ▶ Face stylization 🧑
- Holistic landmark detection

Text tasks

- ▶ Text classification
- ▶ Text embedding
- ▶ Language detection

Audio tasks

- ▶ Audio classification



[Introducing LiteRT Next](#): A new set of APIs that improves and simplifies on-device hardware acceleration.

Home > Google AI Edge > Solutions

Was this helpful?

Pose landmark detection guide

[Send feedback](#)

On this page

- Get Started
- Task details
 - Features
 - Configurations options
- Models
 - Pose landmarker model

The MediaPipe Pose Landmarker task lets you detect landmarks of human bodies in an image or video. You can use this task to identify key body locations, analyze posture, and categorize movements. This task uses machine learning (ML) models that work with single images or video. The task outputs body pose landmarks in image coordinates and in 3-dimensional world coordinates.



Try it! →

ai.google.dev uses cookies to deliver and enhance the quality of its services and to analyze traffic. If you agree, cookies are also used to serve advertising and to personalize the content and advertisements that you see. [Learn more.](#)

Agree

No thanks

Start using this task by following the implementation guide for your target platform. These platform-specific guides walk you through a basic implementation of this task, including a recommended model, and code example with recommended configuration options:

- **Android** - [Code example](#) - [Guide](#)
- **Python** - [Code example](#) - [Guide](#)
- **Web** - [Code example](#) - [Guide](#)

Task details

This section describes the capabilities, inputs, outputs, and configuration options of this task.

Features

- **Input image processing** - Processing includes image rotation, resizing, normalization, and color space conversion.
- **Score threshold** - Filter results based on prediction scores.

Task inputs	Task outputs
The Pose Landmarker accepts an input of one of the following data types: <ul style="list-style-type: none">• Still images• Decoded video frames• Live video feed	The Pose Landmarker outputs the following results: <ul style="list-style-type: none">• Pose landmarks in normalized image coordinates• Pose landmarks in world coordinates• Optional: a segmentation mask for the pose.

Configurations options

This task has the following configuration options:

Option Name	Description	Value Range	Default Value
running_mode	Sets the running mode for the task. There are three modes: IMAGE: The mode for single image inputs. VIDEO: The mode for decoded frames of a video. LIVE_STREAM: The mode for a livestream of input data, such as from a camera. In this mode, resultListener must be called to set up a listener to	{IMAGE, VIDEO, LIVE_STREAM}	IMAGE

Option Name	Description	Value Range	Default Value
	receive results asynchronously.		
<code>num_poses</code>	The maximum number of poses that can be detected by the Pose Landmarker.	Integer > 0	1
<code>min_pose_detection_confidence</code>	The minimum confidence score for the pose detection to be considered successful.	Float [0.0, 1.0]	0.5
<code>min_pose_presence_confidence</code>	The minimum confidence score of pose presence score in the pose landmark detection.	Float [0.0, 1.0]	0.5
<code>min_tracking_confidence</code>	The minimum confidence score for the pose tracking to be considered successful.	Float [0.0, 1.0]	0.5
<code>output_segmentation_masks</code>	Whether Pose Landmarker outputs a segmentation mask for the detected pose.	Boolean	False
<code>result_callback</code>	Sets the result listener to receive the landmarker results asynchronously when Pose Landmarker is in the live stream mode. Can only be used when running mode is set to <code>LIVE_STREAM</code>	ResultListener	N/A

Models

The Pose Landmarker uses a series of models to predict pose landmarks. The first model detects the presence of human bodies within an image frame, and the second model locates landmarks on the bodies.

The following models are packaged together into a downloadable model bundle:

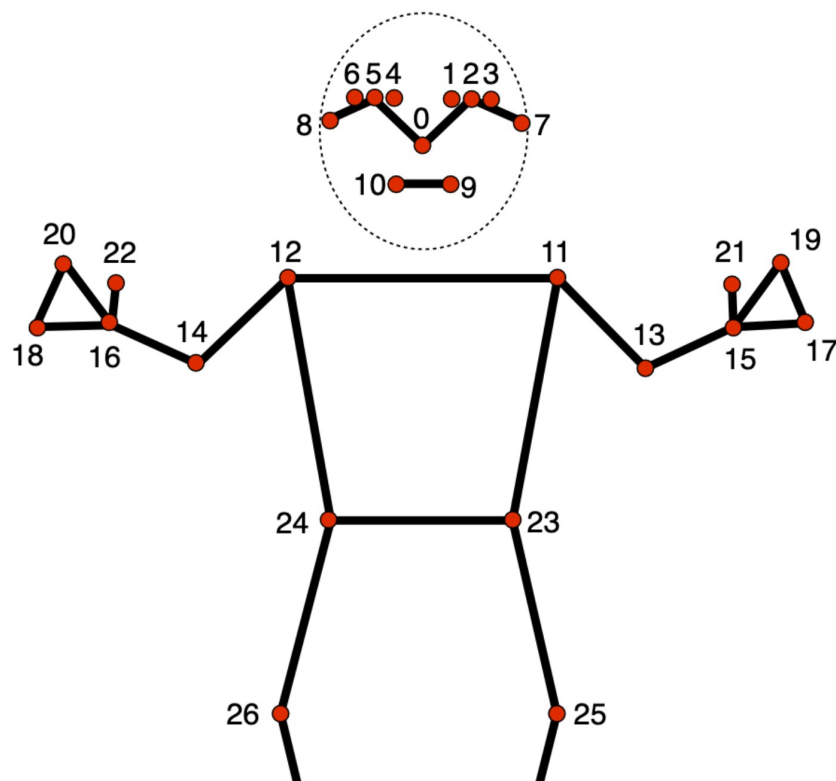
- **Pose detection model:** detects the presence of bodies with a few key pose landmarks.
- **Pose landmarker model:** adds a complete mapping of the pose. The model outputs an estimate of 33 3-dimensional pose landmarks.

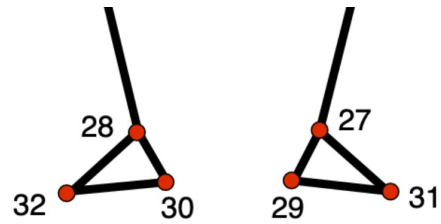
This bundle uses a convolutional neural network similar to [MobileNetV2](#) and is optimized for on-device, real-time fitness applications. This variant of the [BlazePose](#) model uses [GHUM](#), a 3D human shape modeling pipeline, to estimate the full 3D body pose of an individual in images or videos.

Model bundle	Input shape	Data type	Model Cards	Versions
Pose landmark (lite)	Pose detector: 224 x 224 x 3 Pose landmark: 256 x 256 x 3	float 16	info	Latest
Pose landmark (Full)	Pose detector: 224 x 224 x 3 Pose landmark: 256 x 256 x 3	float 16	info	Latest
Pose landmark (Heavy)	Pose detector: 224 x 224 x 3 Pose landmark: 256 x 256 x 3	float 16	info	Latest

Pose landmark model

The pose landmark model tracks 33 body landmark locations, representing the approximate location of the following body parts:





- 0 - nose
- 1 - left eye (inner)
- 2 - left eye
- 3 - left eye (outer)
- 4 - right eye (inner)
- 5 - right eye
- 6 - right eye (outer)
- 7 - left ear
- 8 - right ear
- 9 - mouth (left)
- 10 - mouth (right)
- 11 - left shoulder
- 12 - right shoulder
- 13 - left elbow
- 14 - right elbow
- 15 - left wrist
- 16 - right wrist
- 17 - left pinky
- 18 - right pinky
- 19 - left index
- 20 - right index
- 21 - left thumb
- 22 - right thumb
- 23 - left hip
- 24 - right hip
- 25 - left knee
- 26 - right knee
- 27 - left ankle
- 28 - right ankle
- 29 - left heel
- 30 - right heel
- 31 - left foot index
- 32 - right foot index

The model output contains both normalized coordinates (`Landmarks`) and world coordinates (`WorldLandmarks`) for

each landmark.

Was this helpful?




[Send feedback](#)

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](#), and code samples are licensed under the [Apache 2.0 License](#). For details, see the [Google Developers Site Policies](#). Java is a registered trademark of Oracle and/or its affiliates.

Last updated 2025-01-13 UTC.

[Terms](#) | [Privacy](#) | [Manage cookies](#)

 [Language](#) ▼