

¡Entrena tu propia inteligencia artificial!

Objetivo:

Aprender cómo funciona el reconocimiento de imágenes con inteligencia artificial, entrenando un modelo personalizado y usándolo en una aplicación web con Angular en tiempo real.

Instrucciones:

1. **Accede a Teachable Machine**
Ve a <https://teachablemachine.withgoogle.com/> y selecciona **Image Project > Standard Image Model**.
2. **Crea tus clases**
Añade 2 o 3 clases (por ejemplo: “Yo con gafas”, “Yo sin gafas”, “Mano levantada”). Puedes grabar imágenes con tu webcam directamente.
3. **Entrena el modelo**
Haz clic en **Train Model** y espera a que se complete el entrenamiento.
4. **Exporta el modelo**
 - Haz clic en **Export Model**.
 - Elige la opción **TensorFlow.js**.
 - Copia el **enlace del modelo hospedado** (ejemplo: <https://teachablemachine.withgoogle.com/models/XXXXX/>).

Programación con Angular

Crea el proyecto e instala el módulo de TensorFlowJS:

```
ng new image-classifier-app
cd image-classifier-app
npm install @tensorflow/tfjs
```

Predicción de imágenes con Tensor Flow - Angular

index.html

Agrega esto justo antes de `</body>`:

```
<!-- Cargar primero TensorFlow.js -->
<script
src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs@4.10.0/dist/tf.min.js"></s
cript>

<!-- Luego cargar Teachable Machine -->
<script
src="https://cdn.jsdelivr.net/npm/@teachablemachine/image@0.8/dist/teachablen
achine-image.min.js"></script>
```

A continuación crea un componente llamado **image-classifier** con `ng generate component image-classifier`

image-classifier.component.ts

```
import { Component, ElementRef, ViewChild, OnInit } from '@angular/core';
import { CommonModule } from '@angular/common';
declare const tmImage: any;

@Component({
  selector: 'app-image-classifier',
  standalone: true,
  imports: [CommonModule],
  templateUrl: './image-classifier.component.html',
  styleUrls: ['./image-classifier.component.css']
})
export class ImageClassifierComponent {
  @ViewChild('webcamContainer', { static: true }) webcamContainer!:
  ElementRef;
  model: any;
  webcam: any;
  prediction: string = '';
  isPredicting: boolean = false;

  async ngOnInit() {
    const modelURL =
'https://teachablemachine.withgoogle.com/models/C9zDSXjQn/model.json';
    const metadataURL =
'https://teachablemachine.withgoogle.com/models/C9zDSXjQn/metadata.json';

    this.model = await tmImage.load(modelURL, metadataURL);
```

Predicción de imágenes con Tensor Flow - Angular

```
this.webcam = new tmImage.Webcam(300, 300, true);
await this.webcam.setup();
await this.webcam.play();

this.webcamContainer.nativeElement.appendChild(this.webcam.canvas);

// Iniciar predicción continua
this.isPredicting = true;
this.loop();
}

async loop() {
  if (!this.isPredicting) return;

  await this.webcam.update(); // actualizar frame

  const predictions = await this.model.predict(this.webcam.canvas);
  const top = predictions.sort((a: any, b: any) => b.probability -
a.probability)[0];
  this.prediction = `${top.className} (${(top.probability *
100).toFixed(2)}%)`;

  window.requestAnimationFrame(() => this.loop());
}
}
```

image-classifier.component.html

```
<div class="container">
  <h1>Clasificador de Imágenes IA 🤖</h1>
  <div class="card">
    <div #webcamContainer class="webcam-container"></div>
    <p class="prediction" *ngIf="prediction">🔍 {{ prediction }}</p>
  </div>
</div>
```

image-classifier.component.css

```
.container {
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  margin: 0;
  height: 100vh;
  font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
```

Predicción de imágenes con Tensor Flow - Angular

```
background-color: #f7f9fb;
min-height: 100vh;
}

h1 {
  color: #333;
  margin-bottom: 2rem;
  font-size: 2rem;
}

.card {
  background: white;
  border-radius: 1rem;
  padding: 1.5rem;
  box-shadow: 0 4px 12px rgba(0, 0, 0, 0.1);
  text-align: center;
}

.webcam-container canvas {
  border-radius: 0.5rem;
  border: 2px solid #ddd;
  max-width: 100%;
}

.prediction {
  margin-top: 1rem;
  font-size: 1.2rem;
  color: #2c3e50;
  font-weight: bold;
}
```