

¿Eres de letras o de números?

El **reto** propuesto en esta ocasión es el **reconocimiento de dígitos dibujados (MNIST)**, para lo que usaremos una web en la que los podremos dibujar y darle a un botón de predecir. Las tareas a realizar serán:

1. Entrenar o conseguir un modelo IA para reconocer dígitos (0-9) usando MNIST.
2. Modificar la web existente del experimento anterior para permitir dibujar números en el canvas, limpiarlo, y predecir el número dibujado.

Como no es un curso de web, sino de IA

Parte 1: Entrena y exporta tu modelo IA

A. Entrenamiento del modelo (puede ser en Colab o Jupyter Notebook):

- Usa los datos de MNIST para entrenar un modelo simple (ej. una red neuronal con Keras).
- Exporta el modelo al formato TensorFlow.js.

Referencia:

[Guía oficial para convertir modelos a TensorFlow.js](#)

Te dejo aquí un código de posible entrenamiento del *dataset*. Si quieres probar por tu cuenta hazlo sin mirar este primer código.

```
import tensorflow as tf
from tensorflow import keras
(X_train, y_train), (X_test, y_test) = keras.datasets.mnist.load_data()
X_train, X_test = X_train/255.0, X_test/255.0
model = keras.Sequential([
    keras.layers.Flatten(input_shape=(28,28)),
    keras.layers.Dense(128, activation='relu'),
    keras.layers.Dense(10, activation='softmax')
])
model.compile(optimizer='adam', loss='sparse_categorical_crossentropy',
metrics=['accuracy'])
```

```
model.fit(X_train, y_train, epochs=5)
model.save('mnist_model.h5')
```

Luego convierte el modelo:

```
tensorflowjs_converter --input_format keras mnist_model.h5 ./modelo_mnist/
```

Copia los archivos generados (`model.json` , `group1-shard*`) a la carpeta de tu web.

Parte 2: Modifica la web paso a paso

1. Cambia todos los textos e imágenes

- **Cambia títulos, descripciones y textos:**

Ejemplo:

- `<title>Perros y Gatos</title>` → `<title>Reconocimiento de Dígitos</title>`
 - `<h1 class="display-5 fw-bold">Perros y gatos</h1>` → `<h1 class="display-5 fw-bold">Reconocimiento de Dígitos</h1>`
 - Cambia todas las referencias a "perros y gatos" por "números" o "dígitos".
-

2. Elimina la funcionalidad de la cámara

- **Elimina del HTML:**

- Todo el bloque `<video id="video" ...></video>`
- El botón `<button class="btn btn-primary mb-2" id="cambiar-camara" ...>Cambiar camara</button>`

- **Elimina del JavaScript:**

- Borra o comenta todas las funciones relacionadas con la cámara: `mostrarCamara` , `cambiarCamara` , `procesarCamara` , y las líneas que llaman a estas funciones.
 - Elimina la variable `video` .
-

3. Haz que el usuario pueda dibujar en el canvas

- El canvas principal ya existe:

```
<canvas id="canvas" width="400" height="400" ...></canvas>
```

- **Agrega al JavaScript, después de la declaración de variables, lo siguiente:**

```
// Poner fondo blanco al cargar
ctx.fillStyle = "#fff";
ctx.fillRect(0, 0, canvas.width, canvas.height);

// Dibujo
let dibujando = false;
canvas.addEventListener('mousedown', function(e) {
  dibujando = true;
  ctx.lineWidth = 20;      // Grosor del trazo
  ctx.lineCap = "round";   // Borde redondo
  ctx.strokeStyle = "#000"; // Color negro
  ctx.beginPath();
  ctx.moveTo(e.offsetX, e.offsetY);
});
canvas.addEventListener('mousemove', function(e) {
  if (dibujando) {
    ctx.lineTo(e.offsetX, e.offsetY);
    ctx.stroke();
  }
});
canvas.addEventListener('mouseup', function(e) {
  dibujando = false;
});
canvas.addEventListener('mouseleave', function(e) {
  dibujando = false;
});
```

4. Añade los botones "Limpiar" y "Predecir"

- **Debajo del canvas**, agrega en el HTML:

```
<div class="mb-2">
  <button class="btn btn-secondary" id="limpiar">Limpiar</button>
  <button class="btn btn-primary" id="predecir">Predecir</button>
</div>
```

5. Programa el botón Limpiar

- **Agrega al JavaScript:**

```
document.getElementById('limpiar').onclick = function() {
  ctx.clearRect(0, 0, canvas.width, canvas.height);
  ctx.fillStyle = "#fff";
  ctx.fillRect(0, 0, canvas.width, canvas.height);
  document.getElementById("resultado").innerHTML = "";
};
```

6. Modifica la función de predicción

- Reemplaza la función automática `predecir()` por una función que solo se ejecute al pulsar el botón.
- Modifica el tamaño para que se adapte a MNIST (28×28).
- **Agrega al JavaScript:**

```
document.getElementById('predecir').onclick = function() {
  // Redimensiona el dibujo a 28×28 para el modelo MNIST
  resample_single(canvas, 28, 28, otrocanvas);

  var ctx2 = otrocanvas.getContext("2d");
  var imgData = ctx2.getImageData(0, 0, 28, 28);

  var arr = [];
  var arr28 = [];
  for (var p = 0; p < imgData.data.length; p += 4) {
    var rojo = imgData.data[p] / 255;
    var verde = imgData.data[p + 1] / 255;
    var azul = imgData.data[p + 2] / 255;
```

```

var gris = (rojo + verde + azul) / 3;
arr28.push([gris]);
if (arr28.length == 28) {
    arr.push(arr28);
    arr28 = [];
}
}
arr = [arr];
var tensor = tf.tensor4d(arr);

// ¡Recuerda cambiar la ruta del modelo!
if (modelo != null) {
    var resultado = modelo.predict(tensor).dataSync();
    var predicho = resultado.indexOf(Math.max(...resultado));
    document.getElementById("resultado").innerHTML = predicho;
} else {
    document.getElementById("resultado").innerHTML = "Modelo no cargado";
}
};

```

7. Cambia la carga del modelo

- Asegúrate de que la línea donde cargas el modelo apunte al modelo de dígitos (MNIST):

```

modelo = await tf.loadLayersModel("ruta/model.json");
// Por ejemplo: "modelo_mnist/model.json"

```

8. Elimina cualquier referencia al flujo anterior

- Borra cualquier función, llamada o botón que ya no tenga sentido (cámara, cambiar cámara, procesar cámara, etc.).
- Cambia todos los textos finales y descripciones para que solo hablen de reconocimiento de dígitos.

9. (Opcional) Mejora la experiencia de usuario

¿Te atreves a tocar HTML/CSS?

- Haz que el canvas sea cuadrado y suficientemente grande.
 - Pon borde al canvas para que sea visible.
 - El resultado debe aparecer grande y claro bajo el canvas.
-

Checklist para ver que todo está en su sitio

- ☐ ¿La web permite dibujar con el ratón en el canvas?
- ☐ ¿Hay un botón para limpiar el canvas?
- ☐ ¿Hay un botón para predecir el número dibujado?
- ☐ ¿El resultado aparece correctamente debajo del canvas?
- ☐ ¿Todos los textos hacen referencia a reconocimiento de dígitos?
- ☐ ¿No queda ningún resto del flujo de "perros y gatos" ni de la cámara?
- ☐ ¿El modelo está bien entrenado y cargado correctamente?
- ☐ Si quieres cambiar el logo y los favicon te he dejado los archivos adjuntos a la tarea