

RECONOCIMIENTO DE DORSALES DE JUGADORES DE FÚTBOL

Alberto Martín Martín, Javier Calderón Álvarez, Antonio Saborido Campos

15/12/2023

Resumen	1
1. Planteamiento Teórico	2
2. Implementación	2
Menú:	2
Filtros:	2
Base de Datos:	3
Funciones:	3
Librerías usadas:	3
3. Puesta en marcha	4
En Google Colab:	4
En local:	5
4. Experimentación:	5
Selección de Imágenes:	5
Filtros y Librerías:	6
Procesamiento de Texto:	6
Comparación con Base de Datos:	6
Interfaz de Usuario y Presentación de Resultados:	6
Ejemplos Comentados:	7
5. Conclusiones:	9
6. Tabla de tiempos:	10
7. Bibliografía:	10

Resumen

El proyecto se centra en el desarrollo de un sistema de reconocimiento de jugadores de fútbol utilizando imágenes de sus camisetas como entrada. La idea es aprovechar la información visual de las camisetas, que suelen estar marcadas con el nombre y el número del jugador, para identificar automáticamente a quién pertenece la camiseta. Una vez identificado el jugador, el sistema accede a la base de datos del videojuego EA Sports FC 24 para extraer sus estadísticas.

1. Planteamiento Teórico

- Reconocimiento de Jugadores: El objetivo principal es desarrollar un algoritmo de reconocimiento de jugadores que sea capaz de identificar a quién pertenece una camiseta de fútbol a partir de una imagen de la misma. Esto implica la detección y extracción de texto, como el nombre y el número del jugador, de la imagen de la camiseta.

- Integración con EA Sports FC 24: Una vez que se ha identificado al jugador a través de la camiseta, el sistema debe ser capaz de acceder a la base de datos del videojuego EA Sports FC 24. El objetivo es extraer automáticamente las estadísticas del jugador, como su posición, habilidades, historial de partidos, etc.

- Interfaz de Usuario Amigable: Para hacer que la herramienta sea accesible y útil, se pretende desarrollar una interfaz de usuario utilizando google colab que permita a los usuarios cargar imágenes de camisetas y obtener las estadísticas de los jugadores de forma sencilla.

2. Implementación

La implementación del proyecto se organiza en cuatro bloques fundamentales, cada uno desempeñando un papel crucial en el desarrollo integral del sistema de reconocimiento de jugadores de fútbol a partir de imágenes de camisetas. A continuación, se detallan los aspectos clave de cada bloque:

Menú:

-Interactividad: El bloque de menú se encarga de proporcionar una interfaz interactiva por consola, permitiendo a los usuarios navegar y seleccionar las opciones disponibles de manera intuitiva.

-Selección de Imágenes: Implementa funcionalidades que permiten a los usuarios cargar imágenes de camisetas de fútbol para su posterior análisis.

Filtros:

-Mejora de Imágenes: Este bloque alberga una variedad de filtros diseñados para optimizar la calidad de las imágenes. Estos filtros incluyen la conversión a blanco y negro, resaltado de contornos, mejora de contraste, suavizado y mejora de nitidez. La aplicación de estos filtros es esencial para facilitar la detección y extracción de información, como el nombre y número del jugador.

Base de Datos:

-Comparación y Validación: Aquí se lleva a cabo la comparación entre la información extraída de las imágenes utilizando los filtros y las librerías (easyocr y pytesseract) con la base de datos del videojuego EA Sports FC 24. Este bloque asegura la precisión y validez de la identificación del jugador, garantizando que la información obtenida coincida con los registros existentes.

Funciones:

-Procesamiento de Texto: En este bloque, se implementan las funciones específicas para la detección y extracción de texto de las imágenes. Esto incluye el reconocimiento del nombre y el número del jugador, utilizando las librerías mencionadas (easyocr y pytesseract).

-Acceso a la Base de Datos: Incorporamos funciones que permiten el acceso y la extracción de estadísticas de jugadores desde la base de datos de EA Sports FC 24 una vez que se ha identificado correctamente al jugador a través de la camiseta.

-Presentación de Resultados: Finalmente, desarrollamos funciones para presentar de manera clara y comprensible las estadísticas del jugador obtenidas, ofreciendo una interfaz amigable para los usuarios a través de Google Colab.

La integración efectiva de estos bloques asegura un sistema robusto y completo, capaz de identificar jugadores a partir de imágenes de camisetas, acceder a información relevante en la base de datos y presentar los resultados de manera accesible para el usuario

Librerías usadas:

- **CV2 (OpenCV)**: Se utiliza para cargar imágenes, realizar filtrados y otras operaciones de procesamiento de imágenes.

- **Numpy (np)**: Se utiliza para operaciones matemáticas y manipulación de matrices, especialmente en la manipulación de imágenes.

- **Matplotlib.pyplot (plt)**: Se utiliza para mostrar imágenes y resultados gráficos.

- **Pandas (pd)**: Se utiliza para manejar y analizar datos tabulares, como la lectura de una base de datos de jugadores de fútbol.

- **Pytesseract**: Se utiliza para realizar OCR en imágenes y extraer texto, gracias a esto también imprimimos una imagen señalando las zonas de las cuales extraemos el texto.

- **Easyocr**: Se utiliza para realizar OCR en imágenes y extraer texto y posteriormente pueda ser comparado en la base de datos

3. Puesta en marcha

Tenemos dos opciones para ejecutar nuestro proyecto, contamos con un archivo en formato “ipynb” para que pueda ser ejecutado en google colab y también contamos con varios archivos python junto a un script de instalación de librerías para que pueda ser ejecutado en local.

Todos los archivos necesarios para el funcionamiento se encuentran en este repositorio de github (<https://github.com/Antoniiosc7/TrabajoPID>), y se pueden descargar en el directorio en el que nos encontremos usando el siguiente comando en la terminal:

- “git clone <https://github.com/Antoniiosc7/TrabajoPID.git>”

En Google Colab:

Para poder ejecutar correctamente el código en google colab, necesitaremos crear los directorios con las imágenes que vamos a utilizar, dividiendo las imágenes en tres carpetas según su dificultad de lectura, pudiendo ser fáciles, intermedias o difíciles. Por último también debemos importar el archivo “male players.csv” el cual nos servirá para poder encontrar al jugador al cual pertenece la camiseta y nos mostrará por pantalla más información del mismo.

La estructura final en google colab deberá ser algo parecida a la siguiente foto:

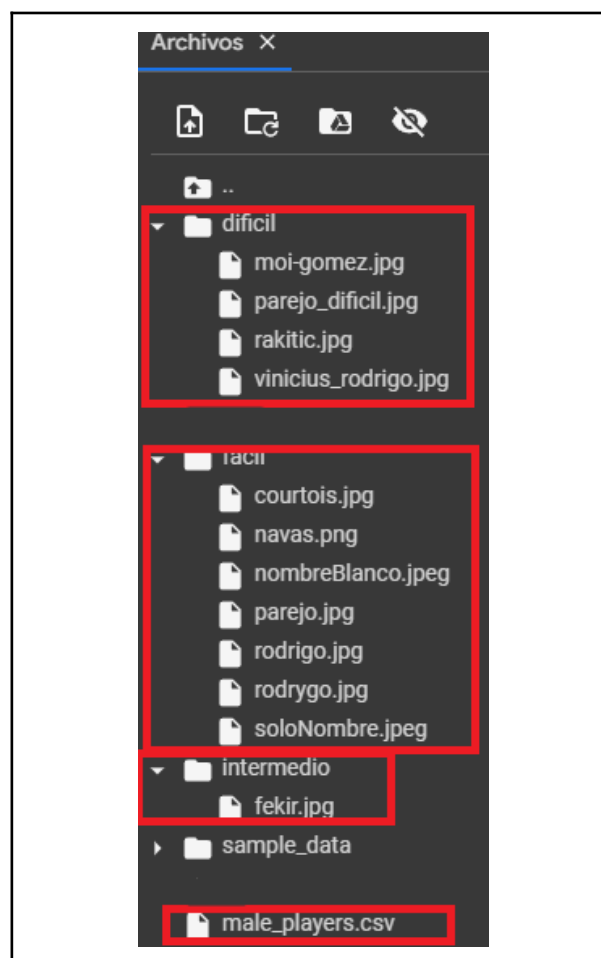


Imagen de ejemplo de la organización en Google Colab

En local:

Para la ejecución en global, una vez hecho el git clone ejecutando el script “*scriptInstalacion.bat*” se deberían de instalar automáticamente todas las librerías usadas para el proyecto. Es probable que aún así falte alguna librería, para su instalación sería “*pip install librería*”.

Una vez instaladas todas las librerías ejecutando el script “*arranque.bat*” ya se debería ejecutar el proyecto y abrirse una terminal así:

Imagen de ejemplo de lo que sucede al ejecutar el script de arranque

Si esta terminal no se abre, se puede hacer desde visual studio code ejecutando el archivo “*mainFile.py*”.

Para añadir nuevas imágenes y probar lecturas tanto en google colab o en local bastará con introducir las nuevas imágenes en algunas de las 3 carpetas disponibles: “*facil*”, “*intermedio*” o “*dificil*”

4. Experimentación:

Selección de Imágenes:

Usamos imágenes de camisetas de jugadores de fútbol con variaciones en el diseño del propio texto de los nombres y los números, y resolución para evaluar la robustez del sistema ante diferentes condiciones.

Filtros y Librerías:

Aplicamos cada uno de los filtros mencionados a las imágenes de las camisetas antes de utilizar las librerías de OCR. Y documentamos los efectos de cada filtro en la calidad de la detección de texto.

Procesamiento de Texto:

Llevamos a cabo pruebas específicas para evaluar la eficacia de las librerías easyocr y pytesseract en la detección y extracción de texto, especialmente en cuanto a la identificación del nombre y número del jugador en las camisetas.

Comparación con Base de Datos:

Realizamos pruebas de comparación entre la información extraída de las imágenes y la base de datos del videojuego EA Sports FC 24. Validamos la consistencia y precisión de los resultados obtenidos.

Interfaz de Usuario y Presentación de Resultados:

Hemos implementado la interfaz de usuario en Google Colab, permitiendo a los usuarios cargar imágenes de camisetas y obtener fácilmente las estadísticas de los jugadores identificados. Realizamos pruebas para evaluar la accesibilidad y la presentación clara de los resultados.

Ejemplos Comentados:

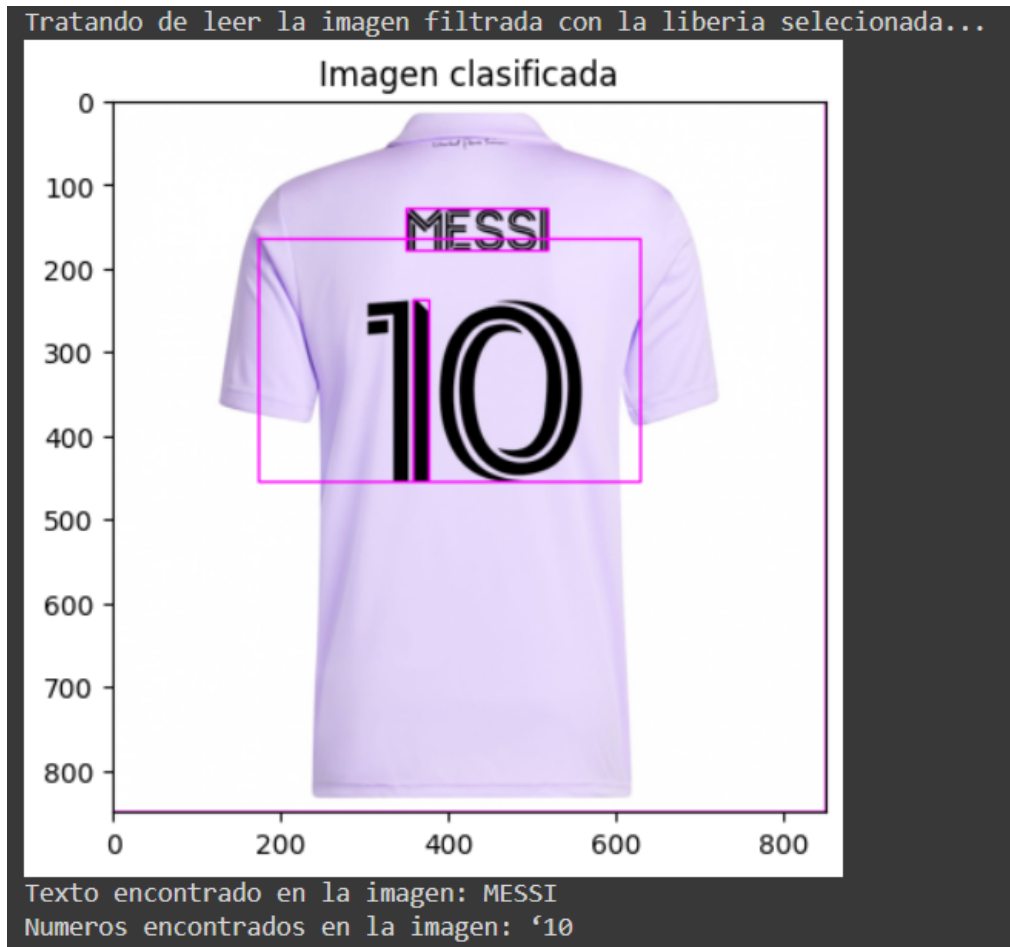
- Ejemplo 1:



Para este ejemplo hemos usado la librería `easyocr` y hemos aplicado el filtro laplaciano. Como podemos observar el texto y el número que encuentra en la imagen aplicando dicha librería y filtro es MESSI y 10.

Una vez extraído el texto y el número aplicaríamos el código respectivo que tenemos para la extracción de las coincidencias de la base de datos, y podemos ver que detecta a dicho jugador y nos proporciona el equipo en el que juega y su nombre completo.

- Ejemplo 2:



Para este ejemplo hemos usado la librería pytesseract y hemos aplicado el filtro contraste. Como podemos observar el texto y el número que encuentra en la imagen aplicando dicha librería y filtro es MESSI y 10.

Gracias a pytesseract podemos sacar como se observa en la imagen que saca los recuadros de la extracción tanto del texto como del número.

Una vez extraído el texto y el número aplicaríamos el código respectivo que tenemos para la extracción de las coincidencias de la base de datos, pero en este caso no nos devuelve nada ya que como se puede observar el número tiene también otro carácter que en este caso es la comilla.

- Ejemplo 3:



Para este ejemplo hemos usado la librería easyocr y hemos aplicado todos los filtros que tenemos. Como podemos observar el texto y el número que encuentra en la imagen aplicando dicha librería y dichos filtros varía dependiendo del filtro escogido.

Como se puede observar cada filtro tiene una precisión distinta a la hora de sacar el número y el texto, por lo tanto no aplicamos el código para buscar la coincidencia con la base de datos.

5. Conclusiones:

En resumen, el proyecto no solo ha logrado el objetivo principal de identificar jugadores a partir de imágenes de camisetas, sino que también ha integrado de manera efectiva tecnologías de procesamiento de imágenes y reconocimiento de texto con una interfaz de usuario intuitiva.

Como conclusión de los resultados obtenidos hemos observado que aplicando dichos filtros hemos obtenido mejores resultados a la hora de leer el texto. Hemos utilizado las librerías de pytesseract y easyocr, donde nos hemos dado cuenta que pytesseract es una librería menos precisa a la hora de devolvernos los resultados, pero hemos comprobado que detecta muy bien los textos gracias al uso de recuadros en la propia imagen. Sin embargo, easyocr tiene una precisión mayor y es la que mejores resultados nos ha proporcionado.

Como conclusión de los resultados esperados, esperábamos una mayor efectividad a la hora de la lectura de texto y de los números de imágenes. Al principio empezamos a usar imágenes de jugadores reales, en las cuales costaba mucho poder leer correctamente dicho texto dado las perspectivas de las imágenes y la cantidad de información externa que nos podríamos encontrar como anuncios, patrocinadores, etc. Por tanto decidimos poner imágenes más simples a poder sin patrocinadores.

Este enfoque puede tener aplicaciones prácticas en diversos entornos, desde el análisis deportivo hasta la creación de herramientas interactivas para los aficionados al fútbol. La combinación de técnicas avanzadas y una interfaz amigable abre nuevas posibilidades para la automatización de tareas relacionadas con la identificación de jugadores y la obtención de información relevante de bases de datos específicas.

Como experiencia de la realización del trabajo, al principio nos agobiamos más ya que no sabíamos usar correctamente las librerías y nos costaba mucho que nos reconociese los resultados obtenidos con la base de datos, pero gracias a ciertos vídeos y documentación pudimos resolverlo.

6. Tabla de tiempos:

Dado a que los tres integrantes del trabajo estamos realizando las prácticas extracurriculares por incompatibilidad horaria no podíamos asistir ir a clase y hacer el seguimiento del trabajo.

7. Bibliografía:

<https://suntrustblog.com/es/text-detection-from-images-using-easyocr-hands-on-guide/>

<https://pythondiario.com/2018/08/extraer-texto-de-imagenes-con-ocr.html>

https://www.youtube.com/watch?v=0OBAApVG4XU&ab_channel=OMES

Y la documentación de la teoría que hay subida en enseñanza virtual.