

Proyecto final: extracción de características en imágenes de minifiguras LEGO

Carolina Maria Burgos Anillo, Edwin Alejandro Caicedo Palacios, William David Moreno Rendón

Repositorio GitHub: https://github.com/cmba-alt/lego_minifigures_features

Video YouTube: https://youtu.be/385D_l8rPfs

Introducción

En el presente proyecto se utiliza el *dataset* de ‘LEGO Minifigures’, disponible en Kaggle [1], para extraer de él 121 imágenes de cuatro clases diferentes, y a partir de estas generar un nuevo *dataset* donde se encuentren características de la distribución del color en las imágenes con sus respectivas etiquetas. Adicionalmente, se realiza un análisis de los momentos de Hu en diferentes imágenes.

Objetivo general

Extraer y analizar características de imágenes del *dataset* de minifiguras LEGO.

Objetivos específicos

1. Extraer la distribución del color en las imágenes a partir de histogramas bidimensionales.
2. Organizar un *dataset* con las características de distribución de color y las etiquetas de la clase correspondiente a cada imagen.
3. Determinar y analizar los momentos de Hu de diferentes figuras.

Desarrollo y resultados

El proyecto se centra en estudiar dos características en imágenes: la distribución del color y los momentos de Hu. Estas se analizarán por separado teniendo en cuenta que la primera se realiza para la generación de un *dataset*, mientras que los momentos se obtendrán solo para su análisis.

1. Características de distribución de colores

El diagrama de bloques de procesos realizados para obtener un *dataset* con características de la distribución de color se presenta en la figura 1.

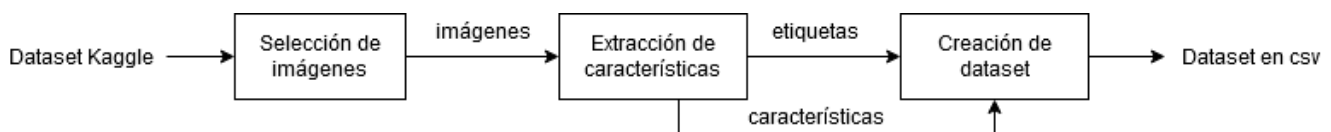


Figura 1. Diagrama de bloques para obtener dataset.

En un principio, al revisar las imágenes contenidas en el dataset, se puede observar que, aunque las figuras tienen colores definidos, es posible que algunas cosas en los fondos no permitan generar buenas características de la imagen, de tal manera que se extrajeron 121 imágenes entre las cuatro clases disponibles, teniendo una distribución como la que se muestra en la tabla 1. Cada una de estas imágenes fue recortada manualmente para quitar el fondo todo lo posible y dejar presente solo la figura del personaje LEGO.

Tabla 1. Número de imágenes por clase.

Clase	Número de imágenes
marvel	31
star-wars	36
jurassic-world	28
harry-potter	26

Para extraer las características, se utilizó la librería de OpenCV para obtener distribución del color en las imágenes a partir de histogramas bidimensionales. Para este fin se utiliza la función `calcHist`, que recibe imágenes en el espacio de color HSV. Este espacio de color se compone de H (Hue) que representa la matiz o tonalidad del color que varía de 0 a 360 grados, la saturación S que se denota como el nivel de saturación del color que varía entre 0 y 1, donde se establece que 0 representa el blanco (Sin saturación presente) y 1 representa el matiz en toda su intensidad. Adicional a lo anterior, V (Value) denota el brillo o intensidad de la iluminación de dicho color, así mismo, 0 representa totalmente oscuro y 1 representa una luminosidad total. En la figura 2 se presenta el modelo y variabilidad de HSV.

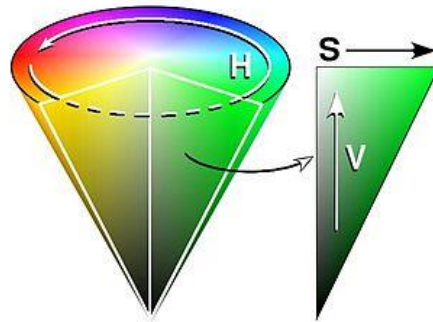


Figura 2. Cono de colores del espacio HSV. [2]

A diferencia de un histograma en 1D, que sólo toma una característica que es el valor de la intensidad del píxel en la escala de grises, el histograma en 2D considera dos características, el valor del Hue y de la saturación S. A continuación, se muestra un ejemplo simple con la primera muestra de las imágenes seleccionadas (imagen 1 carpeta 1 de marvel) para mostrar el histograma 2D con su respectivo valor de Hue (Eje y que va de 0 a 179 por connotación de OpenCV) y valor de saturación (S que va por defecto de 0 a 255).



Figura 3. Imagen Personaje Lego Spiderman.

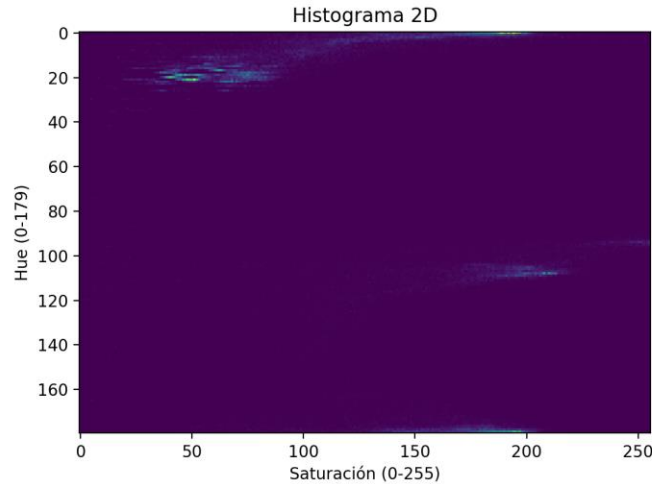


Figura 4. Histograma 2D para la figura 3.

Para tener una mejor vista del umbral que aparece entre 0-50 en su saturación se presenta el gráfico en 3D cómo se evidencia en la siguiente figura:

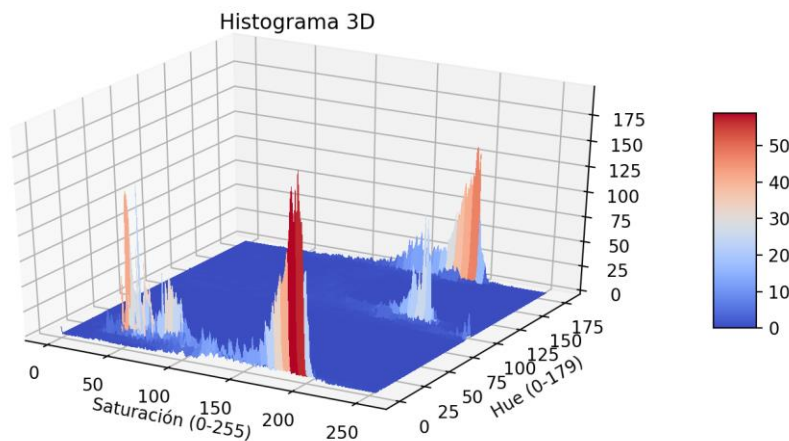


Figura 5. Histograma 3D característico con valores de Hue y Saturación.

Dadas las figuras 3, 4 y 5 se evidencia que la mayoría de los píxeles de la imagen de spiderman es roja y azul con ciertos tonos cafés del fondo, por ende, como se puede apreciar en la figura 4 y 5, los valores de rojo van entre 0-10 y 170-179 en Hue como su variación de 0-255 en saturación que para este caso está entre la mitad y el máximo. Así mismo el color azul que va entre 100-130 de Hue se evidencia en la gráfica.

Ahora bien, es posible escalizar la gráfica de la figura 4 y 5 para tomar en vez de 180 valores de Hue, 16, y en lugar de 256 valores de saturación se toman 4. Con esta relación obtenemos una relación similar al histograma 2D original, pero escalizado, esto a razón de no ocupar tanta memoria porque se guardarían 64 valores en vez de 46080.

El nuevo histograma 2D escalizado se presenta en la figura 6, y el histograma 3D en la figura 7.

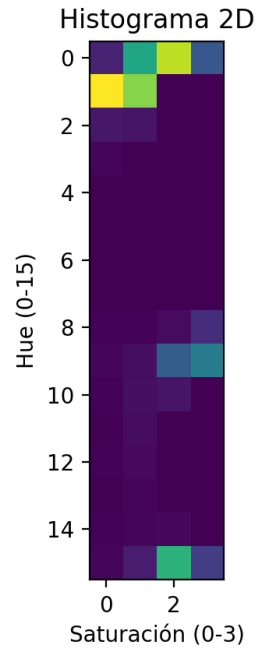


Figura 6. Histograma 2D escalizado.

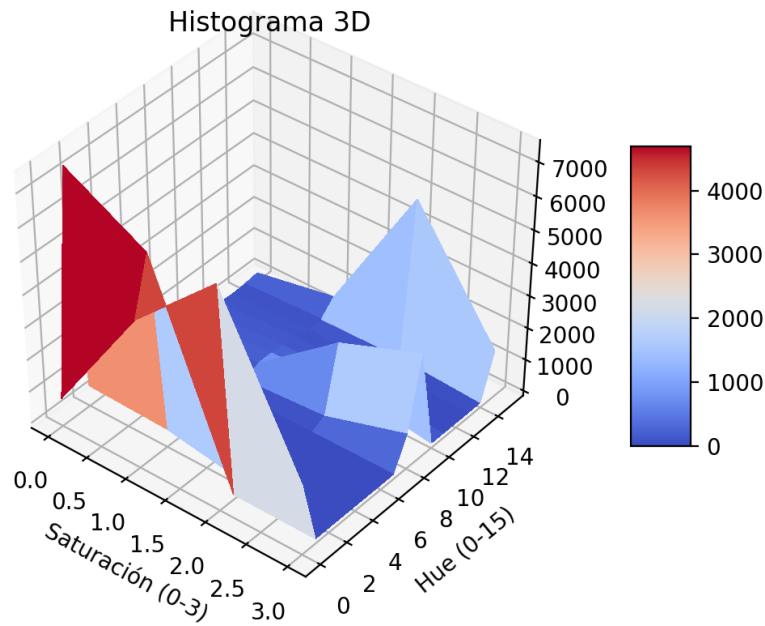


Figura 7. Histograma 3D escalizado.

Para cada imagen de las expuestas en la tabla 1, se obtuvieron los valores del histograma 2D escalizado, con 16 valores para H y 4 para S. Para cada imagen se tomaron los valores de esta matriz de 16x4, y cada uno de los valores individuales representaría una característica en el *dataset* creado. Es decir, se tendrían 64 columnas de características junto a una columna de etiquetas con valores {1, 2, 3, 4} para cada una de las clases, que corresponden a marvel, star-wars, jurassic-world y harry-potter, respectivamente; y el dataset se compone de 121 muestras. El *dataset* se extrae como *lego_data.csv*. Todo este proceso se encuentra en el archivo *datasetg.py*

El proceso realizado para obtener este *dataset* se ilustra en diagrama de flujo de la figura 8.

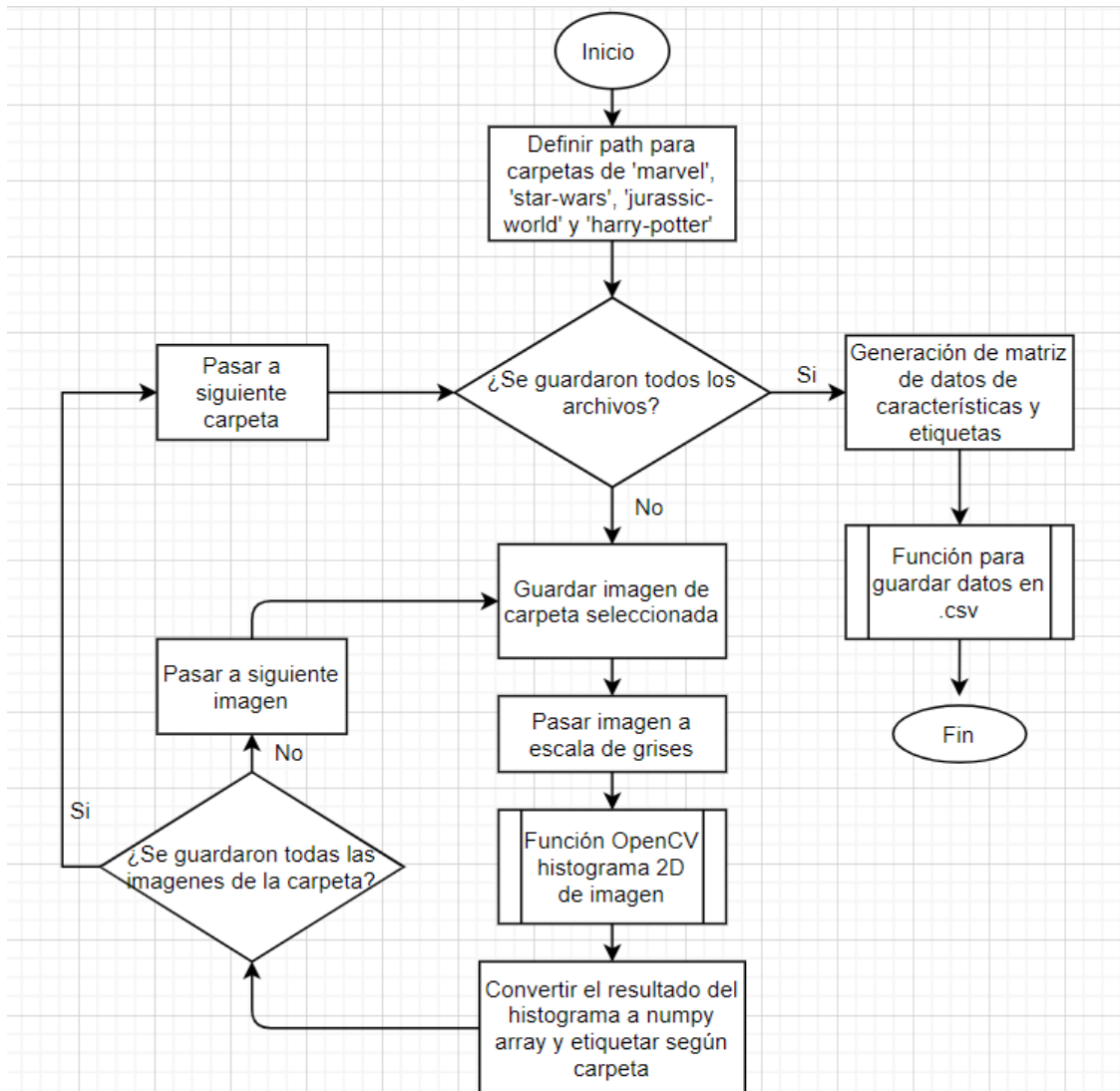


Figura 8. Diagrama de flujo obtención de dataset.

2. Momentos de Hu

Para esta segunda parte del proyecto se quiere analizar un descriptor de una imagen como lo son los momentos de Hu. Los momentos de imagen son un promedio ponderado de la intensidad de los píxeles de una imagen. Los momentos de Hu son invariantes a traslación, rotación y escala [3]. Esto quiere decir que para una figura que se encuentre en una imagen trasladada, rotada o escalada respecto a otra, se obtendrán los mismos valores de momentos de Hu. Son siete momentos, dentro de los cuales los primeros seis son invariantes a las tres operaciones mencionadas y a la reflexión mientras que el séptimo cambia de signo con la reflexión.

OpenCV permite calcular momentos de Hu de manera sencilla. Para esto se realizaron dos experimentos; el primero consiste en calcular los momentos de diferentes imágenes del mismo personaje y evaluar que tanto varían sus momentos, teniendo en cuenta que no es la misma imagen transformada. El segundo

experimento consiste en tomar una sola imagen y realizarle transformaciones para verificar que tanto varían sus momentos, que en teoría debería ser casi nula la variación.

Para el primer experimento se realizó el proceso descrito en el diagrama de flujo de la figura 9.

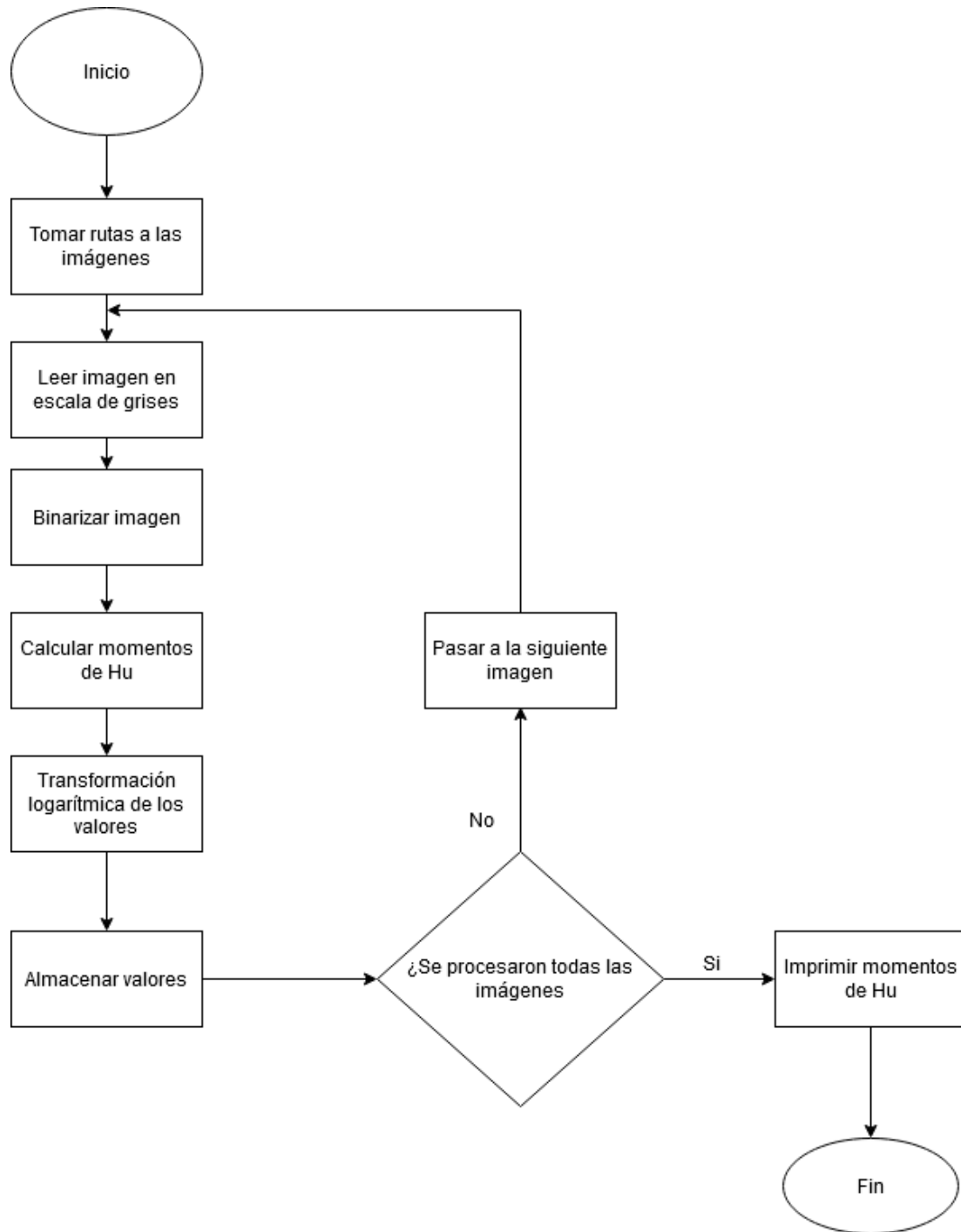


Figura 9. Obtención de momentos de Hu para imágenes de Yoda.

A continuación, se calcularon los momentos de Hu para diferentes imágenes del mismo personaje LEGO en este caso el personaje Yoda de Star Wars. Estas imágenes se encuentran en la carpeta star-wars/0001.

Tabla 2. Momentos de Hu para diferentes imágenes de Yoda.

Imagen.	Momentos de Hu Yoda.						
	H[0]	H[1]	H[2]	H[3]	H[4]	H[5]	H[6]
1	2.713	6.189	9.222	9.275	-19.106	-13.07	-18.54
2	2.902	7.104	10.611	10.233	21.422	13.978	-20.661
3	2.594	7.357	10.708	9.317	19.759	13.057	19.362
4	2.743	7.495	10.054	9.629	-19.719	14.164	19.555
5	2.573	6.211	8.284	9.331	18.172	12.48	18.558
6	2.954	7.849	11.711	10.223	21.199	-14.34	21.898
7	2.719	7.145	9.838	9.903	20.068	13.611	19.838
8	2.816	7.082	9.28	10.143	19.867	13.734	20.502
9	2.954	7.364	11.195	10.706	-21.656	14.392	23.675
10	2.824	7.683	10.18	10.138	20.443	-14.202	20.452
11	2.983	7.342	10.081	12.758	-24.649	16.429	24.203
12	2.954	6.793	11.463	11.482	23.046	15.46	-23.187
Varianza	0.01881919	0.24615	0.910989	0.934248	390.6477	148.1992	329.3451

Como se puede apreciar en la tabla anterior los momentos de Hu tienen valores muy similares en todas las imágenes a pesar de que las imágenes de Yoda LEGO presentan distintas iluminaciones y poses, salvo por algunos valores que son negativos, sin embargo en su valor absoluto siguen siendo cercanos al promedio, se puede apreciar también que la varianza es muy baja para todos los casos exceptuando cuando hay cambio de signo, lo que a manera de aplicación puede ser una buena forma de identificar el objeto, y es que precisamente una de las aplicaciones de los momentos de Hu es reconocer objetos de los cuales previamente se tiene una o varias imágenes con sus momentos calculados previamente [4].

Para el segundo experimento, se sigue el mismo proceso descrito en la figura 9 para encontrar los momentos de Hu, pero esta vez se toma una misma imagen, la correspondiente a star-wars/0001/001.jpg, para la cual se toman esta imagen original, la misma rotada 90° en sentido antihorario, reflejada y escalada, las cuales se pueden observar en la figura 10; y se comparan sus momentos como se realizó en el experimento anterior, los resultados se pueden ver en la tabla 3.

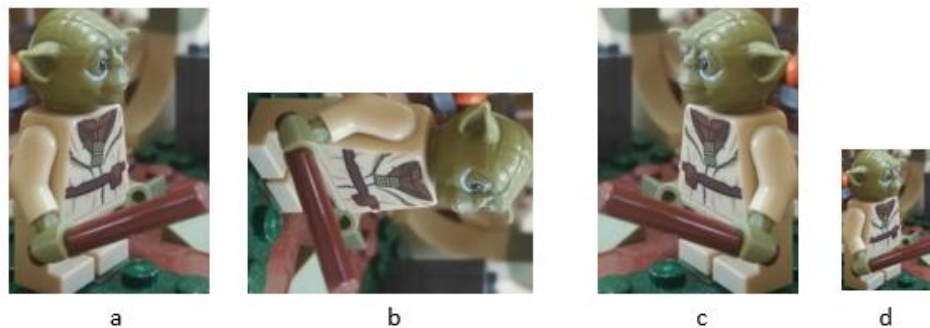


Figura 10. Imagen de Yoda con diferentes transformaciones. a) original. b) rotada. c) reflejada. d) escalada.

Tabla 3. Momentos de Hu para una misma imagen con transformaciones.

	Momentos de Hu yoda.						
IMAGEN	H[0]	H[1]	H[2]	H[3]	H[4]	H[5]	H[6]
original	2.713	6.189	9.222	9.275	-19.106	-13.07	-18.54
rotada	2.713	6.189	9.222	9.275	-19.106	-13.07	-18.54
reflejada	2.713	6.189	9.222	9.275	-19.106	-13.07	18.54
escalada	2.708	6.168	9.175	9.266	-19.001	-12.995	-18.508
varianza	4.69E-06	8.27E-05	4.14E-04	1.52E-05	2.07E-03	1.05E-03	2.58E+02

Se puede apreciar que para la misma imagen con cambios en rotación reflexión y escala los momentos de Hu son casi invariantes, se puede apreciar que la varianza en los distintos casos es baja, (exceptuando los valores negativos), por lo que se puede comprobar como lo dice la teoría que los momentos de Hu son completamente invariantes a estos cambios, por lo que como se mencionó anteriormente pueden ser muy útiles para reconocer objetos cuyos momentos de Hu son bien conocidos, a pesar de que en la escena presente cambios en rotación escala, rotación o reflexión [4].

Conclusiones

A partir de la realización del proyecto fue posible conocer dos diferentes descriptores de características de imágenes. En la primera parte del proyecto, donde se obtuvieron características de la distribución por histogramas bidimensionales, se da cuenta de la utilidad de tener esta información y ordenarla, pues posteriormente puede ser utilizada para tareas de clasificación. También se puede ver que gracias a las funciones que OpenCV tiene disponibles, la extracción de estas características se realiza de manera sencilla. Los histogramas obtenidos, aun en su escala reducida, presentan información suficiente que se puede relacionar con las imágenes de las cuales estos se extraen.

En cuanto a la segunda parte del proyecto, el cálculo de los momentos de Hu es útil en el reconocimiento de objetos cuyos momentos son conocidos previamente, ya que la estimación de los momentos de Hu tiene la ventaja de ser invariante a la escala, la rotación y la traslación que pueda tener el objeto en la escena, pues se pudo apreciar que en diferentes tomas del mismo personaje se obtuvieron momentos de Hu muy similares y con baja varianza. Esto podría considerarse como información útil para clasificación o reconocimiento.

Referencias

- [1] Y. Isaienkov, K. Isaienkov y A. Zhuravel, «LEGO Minifigures Classification,» [En línea]. Available: <https://www.kaggle.com/ihelon/lego-minifigures-classification>. [Último acceso: 2 12 2020].
- [2] Wikipedia, «Modelo de color HSV,» [En línea]. Available: https://es.wikipedia.org/wiki/Modelo_de_color_HSV. [Último acceso: 03 12 2020].
- [3] S. Mallick y K. Bapat, «Shape Matching using Hu Moments (C++/Python),» [En línea]. Available: <https://www.learnopencv.com/shape-matching-using-hu-moments-c-python/>. [Último acceso: 03 12 2020].
- [4] J. Leng, «Analysis of Hu's Moment Invariants on Image Scaling and Rotation,» College of Computer Science, , 2010.