

Proyecto final: Aplicación de técnicas de visión por computador para el entrenamiento de un modelo de reconocimiento de patrones empleando imágenes del juego piedra-papel-tijeras.

Diego Eusse Naranjo, Juan David Ramírez Sánchez, Juan Mauricio Herrera Duran y
Rafael Arturo Matallana Lozano
Universidad Nacional de Colombia – Sede Medellín

Motivación – Actualmente la visión artificial tiene gran importancia por el gran rendimiento que aporta a las diferentes áreas (la industria, la medicina, la seguridad entre otras), ya que mediante la automatización y precisión de las tareas que realiza, puede lograr grandes ahorros en costos de procesos e incluso salvar mayor número de vidas. La principal motivación de realizar este trabajo se encuentra enfocada en aprender a implementar un sistema de visión por computador clásico y aplicar este a la interpretación de imágenes de un caso de un juego de la vida real.

I. PROBLEMA A RESOLVER

Hoy en día la representación de símbolos mediante señas empleando la mano, es un gran tema de interés ya que este representa una forma rápida y alternativa para comunicarse. La fácil captura e identificación de estas señas podría ser de gran utilidad para desarrollar sistemas versátiles que identifiquen los gestos por partes de los humanos. Debido a esto, para realizar un sistema de identificación de gestos en manos, se emplea un ejemplo simple de conocimiento común como lo es el juego de piedras-papel-tijeras, el cual consta de la representación de estos objetos empleando la mano y obedeciendo una secuencia de jerárquica para identificar el ganador de una ronda de dos o más jugadores.

El objetivo del presente trabajo es implementar un sistema de visión por computador que permita interpretar el símbolo que se está representando en la imagen de entrada (Piedra, papel o tijera).

II. SOLUCION PROPUESTA

El sistema de visión por computador clásico fue implementado empleando un dataset de imágenes obtenidas de Kaggle [1], a las cuales se les aplicó las siguientes etapas: adquisición, preprocesamiento, segmentación, selección y extracción de características y reconocimiento de patrones.

El sistema fue construido a través del uso de la herramienta Jupyter, desarrollando cada una de las etapas dentro de un notebook escrito en el lenguaje de programación Python.

III. DESCRIPCIÓN DE LA SOLUCION

A. Adquisición de las imágenes

El dataset de 2188 imágenes utilizado para la implementación del sistema, contiene 3 clases de símbolos, piedra, papel y tijera (**Ilustración 1**), las cuales son representadas de forma simbólica a través de manos reales. La clase piedra contiene un total de 726 imágenes, la clase papel 710 y la clase tijera 752, todas ellas en formato '.png' y en el espacio de colores RGB. Las imágenes obtenidas poseen un fondo común verde para todas las clases, y a su vez una dirección uniforme enfocada hacia la izquierda.



Ilustración 1. Imágenes de muestra del dataset (Piedra, tijera, papel respectivamente)

Empleando la librería de split-folders, se separan los datos en tres conjuntos para su procesamiento, obteniendo 1531 imágenes para entrenamiento, 437 para validación y 220 para pruebas.

B. Preprocesamiento

Dado que hay una diferencia notoria entre las manos y el fondo de las imágenes, se decide aplicar un filtro medianBlur (**Ilustración 2**), el cual permite reducir detalles ruidosos en las manos, como lo son las arrugas o vellos, sin alterar significativamente los contornos.

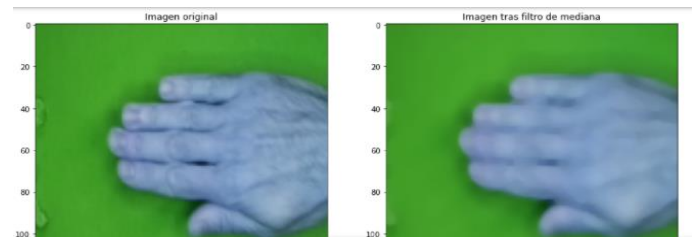


Ilustración 2. Imagen original e imagen tras aplicar filtro medianBlur.

Se logra diferenciar la mano del fondo cambiando las imágenes a un espacio de colores de RGB a HSV, donde se puede apreciar que la mejor opción para esta problemática es elegir el canal H (**Ilustración 3**).

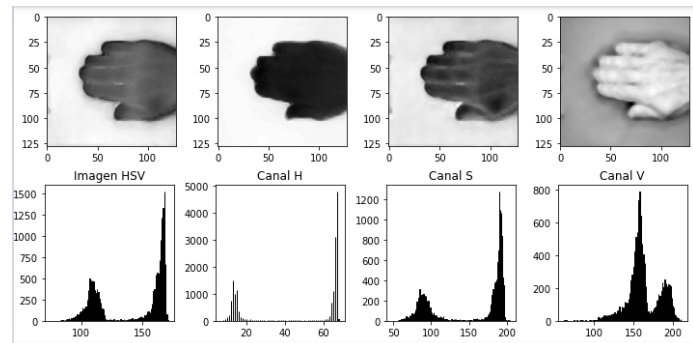


Ilustración 3. Histogramas de capas de espacio de colores HSV.

C. Segmentación

Una vez se tienen las imágenes preprocesadas, se procede a determinar una segmentación adecuada para separar la mano del fondo en un plano binario. Para lo cual, se utiliza el método de segmentación OTSU (**Ilustración 4**) que permite encontrar el umbral de manera automática logrando una segmentación

óptima.

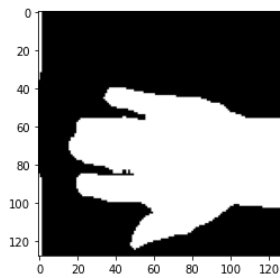


Ilustración 4. Resultado de la aplicación OTSU.

D. Extracción de características

Para la extracción de características que permiten determinar una diferencia entre las imágenes tratadas, se utilizan descriptores de forma, mediante el módulo RegionProps de la librería skimage, la cual permite obtener un diccionario de 20 características extraídas del conjunto de imágenes.

Este conjunto es el insumo para seleccionar las mejores características que permitan reconocer un patrón general de los símbolos representados en las imágenes a evaluar.

E. Reconocimiento de patrones

El reconocimiento de patrones se implementó a través del entrenamiento de 3 modelos de aprendizaje automático: KNN, Random Forest y SVC, cuyos hiper-parámetros fueron ajustados a través de una estrategia de validación cruzada utilizando el conjunto de validación, cada uno de ellos entrenado con 4 características de las seleccionadas en la etapa de extracción de características: X_CENTROID_LOCAL, AREA_PERIMETER_RATIO, COMPACITY y MIN_COL.

En la **Ilustración 5**, se muestra la relación de las dos componentes obtenidas al aplicar el algoritmo PCA sobre las 4 características seleccionadas.

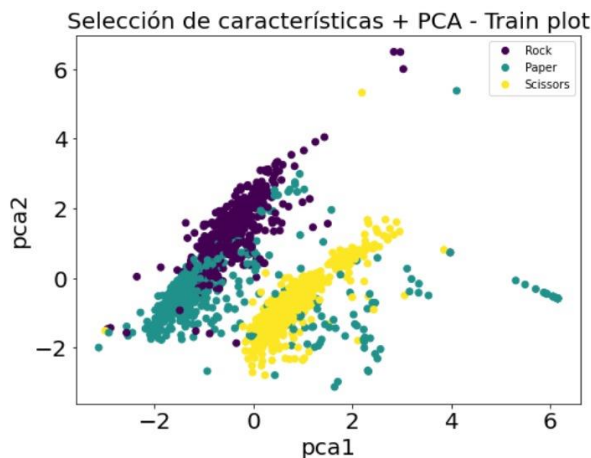


Ilustración 5. Combinación de las 4 características seleccionadas a través de la aplicación del algoritmo PCA para el conjunto de datos de entrenamiento.

Luego de entrenar cada uno de los modelos de reconocimiento de patrones y utilizarlos para clasificar las imágenes del conjunto de pruebas, se obtuvieron las matrices de confusión mostradas en la **Ilustración 6**.

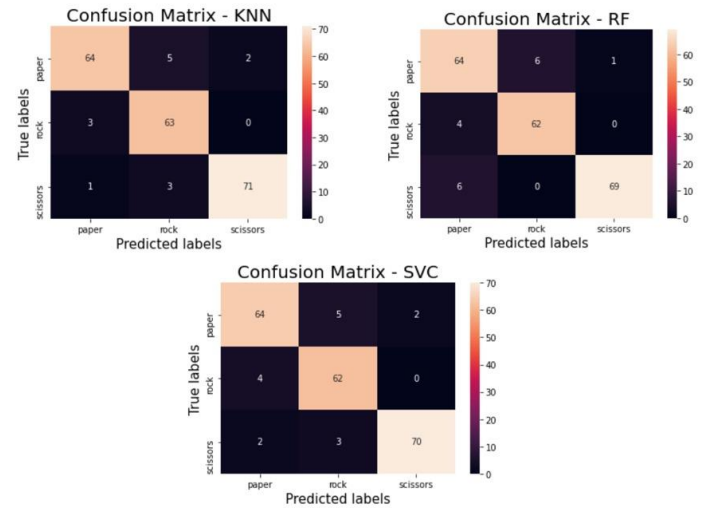


Ilustración 6. Matrix de confusión, modelos KNN, Random Forest y SVC.

F. Análisis de resultados

Como se muestra en la **Ilustración 7**, el algoritmo KNN fue el que mejores métricas presentó, en general, para todas las clases, seguido por el algoritmo SVC y finalmente, por el algoritmo RandomForest.

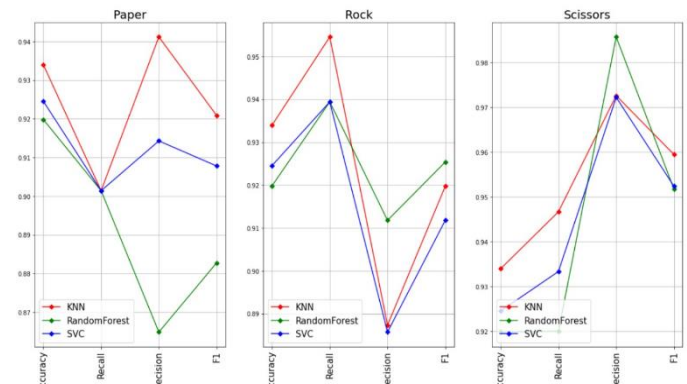


Ilustración 7. Métricas de rendimiento modelos KNN, Random Forest y SVC.

IV. CONCLUSIONES

Se evidenció la importancia de la etapa adquisición de imágenes, ya que inicialmente se trabajó con un conjunto de datos heterogéneo, el cual generaba gran complejidad en la identificación de imágenes por parte de los modelos.

Se evidenció que, para tener una mejor precisión en la clasificación de imágenes, es recomendable la combinación de varias características, ya que con esto los conjuntos de las clases de datos serán más fáciles de generalizar por los modelos a entrenar.

Por medio de este trabajo fue posible evidenciar la importancia de cada uno de los pasos de visión por computador, que permita interpretar el símbolo que se está representando en una imagen de entrada.

V. REFERENCIAS

- [1] Bruère J. (2018). *Rock-Paper-Scissors Images, Images from the Rock-Paper-Scissors game*. Kaggle. Retrieved from <https://www.kaggle.com/drgfreeman/rockpaperscissors>