

Informe - Clasificador de Colores con TCS34725, IQR, EEPROM y KNN en Arduino

Sara Castilla Gualdrón, Jan Escorcía, Jose Quiñonez, Sebastián Sandoval
Universidad de la Costa

Resumen—Este documento presenta el diseño e implementación de un clasificador de colores utilizando el sensor TCS34725 en conjunto con técnicas de procesamiento de datos como el IQR y el algoritmo KNN, integrados en una plataforma Arduino. Se incluyen los fundamentos teóricos, los objetivos del proyecto, el desarrollo experimental, los resultados obtenidos y las conclusiones finales.

I. INTRODUCCIÓN

La clasificación de colores mediante sensores embebidos constituye una herramienta ampliamente utilizada en aplicaciones de robótica, automatización y sistemas de visión artificial de bajo costo. Estos sistemas permiten identificar y diferenciar objetos en función de sus características cromáticas, optimizando tareas industriales, educativas y experimentales. El presente trabajo desarrolla la implementación de un clasificador de colores utilizando el sensor TCS34725, el cual proporciona mediciones crudas de los canales rojo (R), verde (G), azul (B) y un canal de claridad (C) que facilita la normalización de las lecturas. Para mejorar la robustez frente a datos atípicos se emplea el método de filtrado IQR (Interquartile Range), mientras que el almacenamiento de centroides de referencia se gestiona en memoria EEPROM, lo que permite su reutilización sin necesidad de recalibración constante. Finalmente, el algoritmo de clasificación K-Nearest Neighbors (KNN) se encarga de determinar la clase de color a partir de las lecturas procesadas. Este enfoque busca garantizar un equilibrio entre precisión, eficiencia en el uso de memoria y simplicidad de implementación, lo que lo convierte en una alternativa viable para proyectos basados en microcontroladores Arduino orientados al reconocimiento de colores en tiempo real.

II. OBJETIVOS

II-A. Objetivo General

Implementar y documentar un sistema embebido en Arduino que clasifique colores de manera eficiente mediante el uso del sensor TCS34725, técnicas de filtrado IQR, almacenamiento en EEPROM y el algoritmo de clasificación KNN.

II-B. Objetivos Específicos

- Analizar el funcionamiento del sensor TCS34725 y su aplicación en la detección de colores.
- Implementar un filtrado robusto de lecturas utilizando el método IQR para la eliminación de valores atípicos.
- Diseñar un mecanismo de almacenamiento de centroides en EEPROM que optimice el uso de memoria.

- Desarrollar un clasificador basado en el algoritmo KNN para determinar la clase de color de manera rápida y confiable.
- Evaluar experimentalmente el desempeño del sistema en términos de precisión, eficiencia de memoria y tiempo de clasificación.

III. MARCO TEÓRICO

III-A. Sensor TCS34725

El TCS34725 es un sensor de color que entrega valores crudos correspondientes a los canales rojo (R), verde (G), azul (B) y un canal adicional de claridad (C). Este último permite compensar variaciones de iluminación y realizar una normalización de las lecturas. El sensor incluye un conversor analógico-digital (ADC) de 16 bits y un filtro de bloqueo infrarrojo que mejora la precisión en la detección de colores. Su comunicación se realiza mediante el protocolo I²C, lo que facilita su integración con plataformas como Arduino.

III-B. Filtrado IQR (Interquartile Range)

El IQR es una técnica estadística utilizada para eliminar valores atípicos en un conjunto de datos. Se calcula como la diferencia entre el tercer cuartil (Q3) y el primer cuartil (Q1):

$$IQR = Q3 - Q1 \quad (1)$$

Con estos valores se determinan los límites aceptables:

$$Lower = Q1 - 1,5IQR, Upper = Q3 + 1,5IQR \quad (2)$$

Cualquier muestra que se encuentre fuera de este rango se considera un outlier y se descarta. En este proyecto, la técnica IQR se aplica a múltiples lecturas del sensor TCS34725 con el fin de obtener valores más estables y representativos de cada color.

III-C. Memoria EEPROM

La EEPROM (Electrically Erasable Programmable Read-Only Memory) es una memoria no volátil integrada en muchos microcontroladores. Su característica principal es que permite almacenar datos de forma permanente, incluso cuando el sistema se apaga. En este proyecto se utiliza para guardar los centroides de cada clase de color previamente calculados, evitando que el sistema deba recalibrarse en cada reinicio. Esto optimiza el uso de recursos y garantiza una clasificación consistente en diferentes sesiones de operación.

III-D. Algoritmo K-Nearest Neighbors (KNN)

El KNN es un algoritmo de clasificación supervisada basado en la distancia entre un punto de entrada y un conjunto de muestras previamente almacenadas. Para determinar la clase de un color, se calculan las distancias entre la muestra actual y los centroides guardados en la EEPROM. En este proyecto se emplea la distancia euclidiana:

$$d(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2} \quad (3)$$

Donde x corresponde a la muestra a clasificar y y al centroide de referencia (R, G, B). El sistema selecciona la clase con la menor distancia, asignando la etiqueta correspondiente al color detectado.

IV. RESULTADOS

Durante la fase experimental se llevaron a cabo tres etapas: entrenamiento de clases, almacenamiento de centroides en memoria EEPROM y clasificación en tiempo real.

IV-A. Entrenamiento y Cálculo de Centroides

En la primera etapa se entrenaron 20 muestras por clase de color. Para la clase ROJO, el sistema almacenó un total de 20 lecturas, generando un centroide promedio de:

$$\text{Centroide Rojo} = (255,00, 255,00, 255,00) \quad (4)$$

De forma similar, para la clase VERDE se registraron 20 muestras, obteniendo como centroide:

$$\text{Centroide Verde} = (175,60, 48,85, 41,00) \quad (5)$$

Estos centroides se almacenaron en la memoria EEPROM, permitiendo su reutilización en posteriores clasificaciones sin necesidad de reentrenar el sistema.

IV-B. Clasificación en Tiempo Real

En la segunda etapa, el sistema se sometió a pruebas con diferentes combinaciones de valores RGB. En todos los casos, el clasificador KNN logró asignar correctamente la clase correspondiente con probabilidades del 100

Ejemplos representativos incluyen:

- Para la entrada (173,55,30), el sistema clasificó como NARANJA con probabilidad del 100
- Para la entrada (76, 128, 46), se predijo VERDE con probabilidad del 100
- Para la entrada (178, 48, 41), el sistema identificó ROJO con probabilidad del 100

El tiempo promedio de procesamiento por clasificación fue de 1–2 ms, evidenciando la eficiencia del algoritmo en un entorno embebido de recursos limitados.

IV-C. Interpretación

Los resultados obtenidos demuestran que el sistema es capaz de:

- Diferenciar de manera precisa entre las clases de color Rojo, Verde y Naranja.
- Garantizar robustez gracias al filtrado de valores atípicos mediante IQR.
- Conservar los centroides en memoria EEPROM, evitando la recalibración en cada inicio.
- Clasificar en tiempo real con tiempos de respuesta menores a 2 ms, lo cual lo hace viable para aplicaciones de visión en sistemas embebidos de bajo costo.

En general, la precisión del clasificador alcanzó el 100

Cuadro I
RESULTADOS DE CLASIFICACIÓN EN TIEMPO REAL

R	G	B	Predicción	P(Rojo)	P(Naranja)	P(Verde)	Tiempo (ms)
173	55	30	Naranja	0.00 %	100.00 %	0.00 %	2
169	57	32	Naranja	0.00 %	100.00 %	0.00 %	2
71	119	58	Verde	0.00 %	0.00 %	100.00 %	2
76	128	46	Verde	0.00 %	0.00 %	100.00 %	1
132	68	57	Rojo	100.00 %	0.00 %	0.00 %	2
178	48	41	Rojo	100.00 %	0.00 %	0.00 %	2
133	69	56	Rojo	100.00 %	0.00 %	0.00 %	2

En los resultados obtenidos se observa que el sistema de clasificación logra identificar correctamente las tres clases de colores entrenadas (Rojo, Naranja y Verde) con un nivel de confianza del 100

V. CONCLUSIÓN

El desarrollo e implementación del clasificador de colores basado en el sensor TCS34725, junto con las técnicas de filtrado mediante IQR, almacenamiento en EEPROM y clasificación por medio de KNN, permitió obtener un sistema eficiente, preciso y con bajo consumo de recursos. Los resultados experimentales demostraron una correcta identificación de las clases de colores Rojo, Naranja y Verde, alcanzando altos niveles de confiabilidad en las predicciones y tiempos de respuesta en el orden de los milisegundos, lo cual valida su aplicabilidad en entornos de tiempo real. Asimismo, se comprobó que la integración de técnicas estadísticas y algoritmos de clasificación optimiza el desempeño del microcontrolador, constituyendo una alternativa práctica y de bajo costo para sistemas de visión embebidos.