

# Juego Didáctico Para Niños Utilizando OpenCv - Python.

Edison Chavez

Ingeniería en Sistemas Informáticos y Computación  
Universidad Técnica Particular de Loja  
El Pedestal, Loja, Ecuador  
Email: eechavez2@utpl.edu.ec

Roddy Yaguana

Ingeniería en Sistemas Informáticos y Computación  
Universidad Técnica Particular de Loja  
Los Geraneos, Loja, Ecuador  
Email: rsyaguana@utpl.edu.ec

**Resumen**—La evolución de la tecnología día a día avanza buscando mayor comodidad para los usuarios, facilitando el vivir diario, permitiendo reducir tiempos, costos y recursos, el avance de la tecnología permite mayor aprendizaje.

En este documento se presentara un juego didactico utilizando herramientas como: OpenCv la librería para poder hacer el reconocimiento de imágenes y otras funciones. Vamos a utilizar como lenguaje de programación python que tiene como función vincularse con la librería de OpenCv y así poder realizar de la mejor manera nuestra aplicación. La finalidad de este documento es presentar un juego para niños entre 5 y 7 años para poder resolver operaciones aritméticas como son las tablas de multiplicar.

**Keywords**—Palabras Reservadas: multiplicar; Opencv; python.

## I. INTRODUCCIÓN

La visión artificial la podemos definir como un campo de la inteligencia artificial, utilizando adecuadamente sus herramientas podemos procesar imágenes digitales para obtener información detallada de ellas.

OpenCv es una librería open source para software de visión artificial ya que esta provee una infraestructura para aplicaciones de visión artificial.

Se utiliza para la detección de personas en vídeos y en imágenes digitales para darles un mejor realce, OpenCV [9] está escrito en C++, tiene interfaces en C++, C, Python [8], Java y MATLAB interfaces y funciona en Windows, Linux, Android y Mac OS.

La detección de figuras se va a utilizar en el presente trabajo para que el estudiante o cualquiera que utilice nuestra aplicación mediante un objeto pueda seleccionar la respuesta correcta de la multiplicación.

## II. DESARROLLO

Para el seguimiento de objetos, se implementara el algoritmo de CamShift que se caracteriza por la actualización en caso de sufrir modificaciones en el tiempo de procesamiento. Como modelo de color se utiliza es el HSV, matriz (Hue), saturación (saturation) y luminancia (value) este espacio es ampliado utilizado en aplicaciones de procesamiento de imágenes [5].

### A. Algoritmo de Camshift

El algoritmo necesita tres partes para que funcione correctamente como se ve en la 1.

**Inicialización:** Se adquiere la imagen inicial, esta es transformada

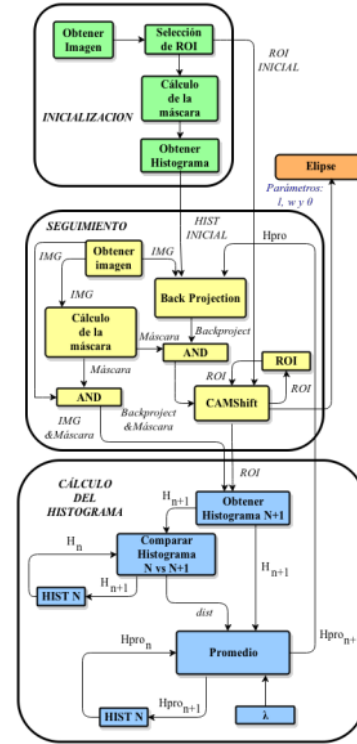


Figura 1. Fases del Algoritmo CamShift [6]

al modelo de color elegido (HSV)[1] y se selecciona la region de interes (ROI) con dicha region se calcula el histograma patron incial, eliminando la informacion del color de fondo. En esta etapa se define el bjeeto a seguir para aquello que se toma una imagen inicial en la cual se define manulamente la ROI, ecuacion 1:

### B. Algoritmo de Camshift

$$\left\{ \hat{P}_v = \min\left(\frac{255}{\max(\hat{q})} \hat{q} v' - 255\right) \right\} u + 1 \dots m \quad (1)$$

Donde:

- $\hat{P}u$ . Representa el valor del histograma, calculado a partir de la ecuación
- $q'$ . Representa los valores binarios del histograma que escalen desde 0 hasta el valor maximo(rango 0 - 255)

$$\left\{ \hat{q}v = \sum_{i=0}^n \sigma[c(x_*^i) - u] \right\} [6] \quad (2)$$

Donde:

- $\hat{q}v$  Representa el valor obtenido del algoritmo de retroproyección del histograma [3]
- $q'$ . Representa los valores binarios del histograma que escalen desde 0 hasta el valor maximo(rango 0 - 255)
- $(x_*^i)$ . Representa la localización de los pixeles del area de interes (ROI) en la imagen.
- $\sum_{i=0}^n$ . Representa la sumatoria de los valores obtenidos en el histograma.
- $\sigma[c... - u]$ . Asocia la localización con el valor del histograma.

#### Seguimiento:

Se ubica el objeto en seguimiento en un objeto o en un frame.

#### Actualización del Histograma:

Con el resultado anterior se actualiza el patron para incorporar las variaciones del brillo y el color del objeto en seguimiento [4].

### III. RETROPROYECCIÓN DEL HISTOGRAMA

Es una operacion primitiva que asocia los valores de los pixeles en la imagen con el valor del bin de histograma correspondiente. la retroproyeccion del histograma de destino con cualquier trama consecutiva gen era una imagen de probabilidad donde el valor de cada pixel caracteriza la probabilidad de que el pixel de entrada pertenezca al histograma que se utilizo[3]. la ecuación 2 muestra como calcular la retroproyección del Histograma

#### A. Trabajos Relacionados

La vision por computador tiene como objetivo resolver varios de los problemas que se presentan en la vida cotidiana de los seres humanos. Todo lo que los humanos podemos observar es a travez de nuestra vision, es por esto que hemos investigado varios proyectos para poder captar ideas de como funciona la vision por computador a traves de la deteccion de movimiento de objetos, y así a travez de nuestro juego darles un aprendizaje mas interactivo a los estudiantes.

**Deteccion de Objetos:** Consiste en la deteccion de objetos de color amarillo, en el se muestra la funcionalidad de como va a detectar el objeto y que efecto va a tener en el mismo [7]. Este proyecto fue de mucha ayuda ya que en el se explica como se hace la deteccion de movimientos a traves de objetos de colores, en nuestro caso utilizaremos el color amarillo.

**Juego de deteccion de movimiento:** Este juego consta de una interfaz y un buen desarrollo del manejo de los objetos. El juego consiste en manejar dos marcadores que funcionan como dos objetos, estos van a permitir manipular a un personaje de

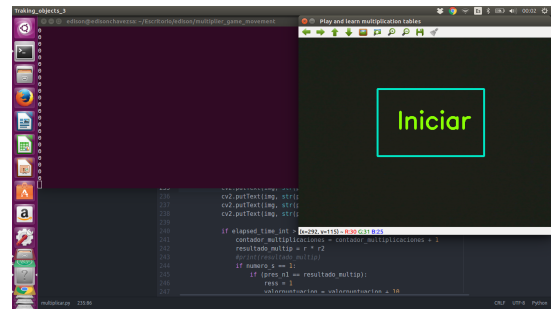


Figura 2. Inicio del Juego

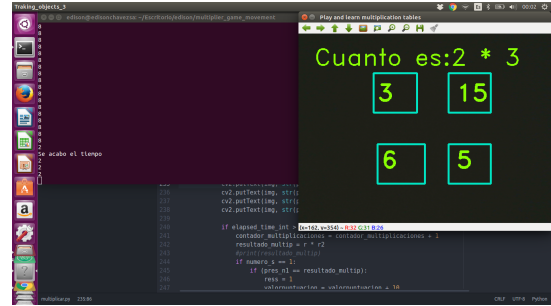


Figura 3. Operaciones a Resolver

un lado a otro [2].

Al revisar el juego ya mencionado nos sirvio para ver como nuestro juego tiene que interactuar con el estudiante, para que de esta manera sea mas interactivo y el estudiante le preste mas atencion.

### IV. RESULTADOS

El juego a realizar tendra que realizarl el usuario las siguientes interacciones:

1. El usuario con el objeto en seguimiento tendra que mantener 3 segundos en empezar para comenzar el juego tal y como se ve en la figura 2.
2. Comenzará el juego presentandose las operaciones y los resultados que puede seleccionar el usuario, ver figura 3.
3. El usuario con el objeto en seguimiento para poder seleccionar la respuesta debera ir con el objeto y esperar 3 seg en la respuesta que desea seleccionar
4. Una vez seleccionada la respuesta, presentara un

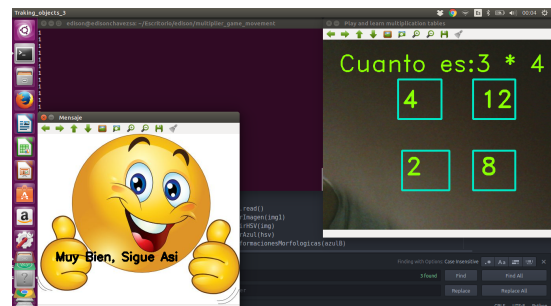


Figura 4. Mensaje de respuesta correcta

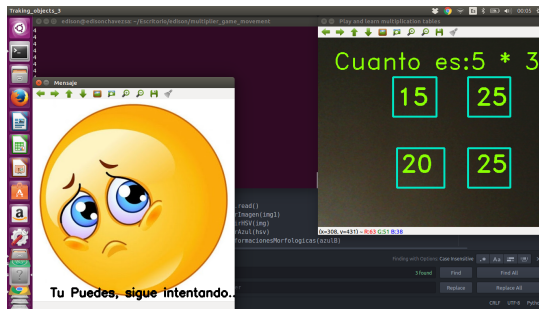


Figura 5. Mensaje de respuesta incorrecta

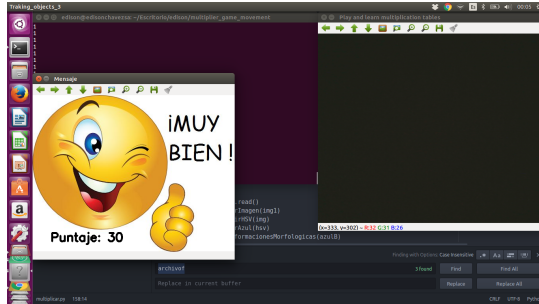


Figura 6. Puntaje Obtenido en el Juego

mensaje de si fue correcto, figura 4 o no figura 5 lo que selecciono.

5. Saldrán 5 operaciones que deba seleccionar
6. Una vez terminado de jugar, saldrá el puntaje acumulado, es decir las respuestas correctas que ha seleccionado, figura 6.

## V. CONCLUSIONES

- El algoritmo camshift permite el seguimiento de objetos a partir de una reconstrucción del histograma.
- Camshift es un algoritmo fácil de implementar y lograr identificar objetos en tiempo real de cualquier tamaño y color.
- El rendimiento del algoritmo, se observa que detecta el objeto y sigue continuamente hasta en oclusiones cortas, de la manera que no deje la zona de búsqueda logrando una mejor segmentación del objeto y disminución del ruido.
- Para que el juego tenga mejores resultados tenemos que estar en un ambiente de luz controlado, para así facilitar la detección del objeto.
- El juego tiene que seguir los principios de openCv ser libre para que otras personas en un futuro puedan modificarlo y así poderlo mejorar, o en su defecto poderlo descargar y así poderlos implementar en instituciones educativas.

## REFERENCIAS

- [1] Aditi Jog. Multiple Objects Tracking using CAMshift Algorithm in Open CV . url:<http://www.iosrjournals.org/iosr-jvlsi/papers/vol1-issue2/G0124146.pdf?id=1959>, 2015.

- [2] Alberto Escudero Pardo. RECONOCIMIENTO DE OBJETOS. url:<http://www.maia.ub.es/~sergio/linked/alberto09.pdf>, 2009.
- [3] D.Comaniciu P. Mean shift:A robust approach toward feature space analysis . url:<https://courses.csail.mit.edu/6.869/handouts/PAMIMeanshift.pdf>, 2002.
- [4] Konik H. Hidayatullah, P. CAMSHIFT Improvement on Multi-Hue and Multi-Object Tracking. In: *International Conference on Electrical Engineering and Informatics*. Bandung, Indonesia (2011), 2012.
- [5] John G. Allen. Object Tracking Using CamShift Algorithm and Multiple Quantized Feature Spaces . url:<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.90.4741rep=rep1type=pdf>, 2006.
- [6] M. Prieto,M. Marufo. Algoritmo de seguimiento de objetos mediante reconstrucción iterativa de histograma en tiempo real . url:[https://www.researchgate.net/profile/Leandro\\_DiMatteo/publication/26999437-de-seguimiento-de-objetos-en-imagenes-mediante-reconstruccion-iterativa-de-histograma-en-tiempo-real.pdf](https://www.researchgate.net/profile/Leandro_DiMatteo/publication/26999437-de-seguimiento-de-objetos-en-imagenes-mediante-reconstruccion-iterativa-de-histograma-en-tiempo-real.pdf), 2014.
- [7] Paul walker. Motion Detection Platform Game Using OpenCV and Python. url:<https://tune.pk/video/3708232/motion-detection-platform-game-using-opencv-and-python>, 2015.
- [8] Python 2.7.13 documentation. Python 2.7.13 documentation. url:<https://docs.python.org/2/>, 2017.
- [9] S.V. Viraktamath,Mukund Katti,Aditya Khatawkar, Pavan Kulkarni. Face Detection and Tracking using OpenCV . url:<http://www.thesij.com/papers/CNCE/2013/July-August/CNCE-0103540102.pdf>, 2016.