



UNIVERSIDAD CENTRAL DEL ECUADOR

Facultad de Ciencias Físicas y Matemática

Ingeniería en Computación Gráfica

Detección, Reconocimiento y Censura de Polígonos Regulares

Josselyn Chaquinga, Juan Carlos Cuaycal, Luis Dominguez, Michael Sarmiento, Martín Oña
November 15, 2018

Abstract

Este proyecto presenta los métodos que se utilizan para la detección de diferentes figuras geométricas como círculos, cuadrados y triángulos, utilizando la transformada de Hough y la Extracción de Características.

Además se describirá y analizará el funcionamiento de estos métodos y el proceso que se sigue, mediante la implementación de diversos parámetros como área, bordes, radio, etc, que complementarán el procedimiento de la detección de figuras geométricas.

Palabras Clave: Visión Computacional, Censura de Imágenes, Detección de Formas, Detección de Bordes, Figuras Geométricas, Transformada de Hough, Extracción de Características.

1 Introducción

Uno de los campos de investigación muy interesante y que esta en creciente auge en diversas aplicaciones, es la Visión Computacional, la misma que describe la deducción automática de la estructura y propiedades del mundo tridimensional a partir de una o varias imágenes bidimensionales del mundo. [1]. Existen diferentes métodos para detectar figuras geométricas básicas como el círculo, el cuadrado o un triángulo, estos métodos son como la transformada de Wavelet, Hough, Algoritmos genéticos, algoritmo de EMO, extracción de características, etc. Los diferentes métodos pueden ser aplicados en algunos lenguaje de programación e implementando diferentes librerías como Python, OpenCV, Java, Matlab,

etc, para el desarrollo de este proyecto se utilizará uno de los lenguajes más conocidos como lo es Python y la librería OpenCV. Los polígonos regulares, como los cuadrados, triángulos y círculos serán detectados y reconocidos, utilizando un dataset de elaboración propia, en el cual se encontraran estas figuras geométricas y luego serán censuradas, censurar es bloquear una forma o figura en particular para comprobar que la figura fue reconocida correctamente. [5] La transformada de Hough, esta técnica se utiliza para detectar figuras geométricas, principalmente detecta líneas rectas, y curvas como círculos y elipses. [2] El proyecto constará con parámetros específicos para su detección como el radio máximo, el radio mínimo y el gradiente. La extracción de características, se en-



carga de obtener las mejores características que representen y describan matemáticamente a las figuras geométricas como es los vértices, bordes y área.[4]

2 Objetivo General

Desarrollar un proyecto en Python y OpenCV que detecte, reconozca y censure polígonos regulares como cuadrados, círculos y triángulos, gracias a la implementación de métodos como la extracción de características y la transformada de Hough.

2.1 Objetivos Específicos

Los Objetivos Específicos, ayudarán a cumplir con el Objetivo General del proyecto que se esta desarrollando.

2.1.1 Implementar la detección y reconocimiento de polígonos regulares en el lenguaje de programación Python y utilizando la librería OpenCV.

2.1.2 Implementar un procedimiento adecuado a un dataset de imágenes de polígonos regulares de tres y cuatro lados, para extraer sus características y lograr la detección de las figuras geométricas.

2.1.3 Detectar círculos mediante el método de la Transformada de Hough, que tendrá como parámetros específicos la imagen, la función del gradiente en los bordes, acumulador de imagen, la mínima distancia, parámetro 1, parámetro 2, el mínimo radio y el máximo radio.

3 Alcance del Proyecto

El proyecto se centra en detectar, reconocer y censurar figuras geométricas tales como círculos, cuadrados y triángulos mediante métodos como

la extracción de características y la transformada de Hough.

3.1 Se desarrollará el aporte de la creación de un dataset de imágenes de polígonos regulares como son cuadrados y triángulos, mediante la extracción de características de los vértices, bordes y el área, luego se procederá a aplicar un censurado a las figuras detectadas y se mostrará por pantalla.

3.2 Se implementará la Transformada Hough para la detección de círculos, en un dataset de imágenes que contienen esta figura, este método contará con parámetros fijos como son el radio máximo, radio mínimo y el gradiente, luego la imagen será censurada y se mostrará por pantalla los círculos censurados y que cumplan con las condiciones planteadas.

3.3 Por último, se realizará un análisis de los resultados obtenidos, gracias a las pruebas que se realizó con el dataset, que contenía 100 imágenes y así se verificará la eficiencia y funcionalidad del sistema.

4 Actividades Desarrolladas

El presente proyecto fue realizado en el lenguaje de programación Python y la librería OpenCV y cuenta con tres partes:

- Detección de Cuadrados-Rectángulos
- Detección de Triángulos
- Detección de Círculos

4.1 Detección de Cuadrados- Rectángulos

En esta clase se crea un método , encargado de extraer todo los contornos del dataset de imágenes, el mismo que será usado más adelante. Primero se lee el dataset, que consta de 100 imágenes, cada imagen consta con figuras básicas como cuadrados, círculos, triángulos y otras figuras adicionales, una vez que se lee las imágenes, se procede a extraer todos los contornos, éstos contornos estan formados por puntos que son almacenados en un array, luego por medio de la extracción de características , en este caso de vértices, se procede a dibujar los contornos de que solo formen cuadrados o rectángulos, es decir aquellos que tengan solo cuatro vértices. Por último se realiza la censura, la misma que se representa de color negro y que esta delimitada por el contorno de los cuadrados y rectángulos que fueron detectados. La detección de cuadrados y rectángulos posee ciertas limitaciones, como por ejemplo solo se detectan cuadrados y rectángulos bien dibujados y sin ningún tipo de rotación En la figura 1 se puede apreciar los resultados que se obtuvo en las imágenes del dataset.

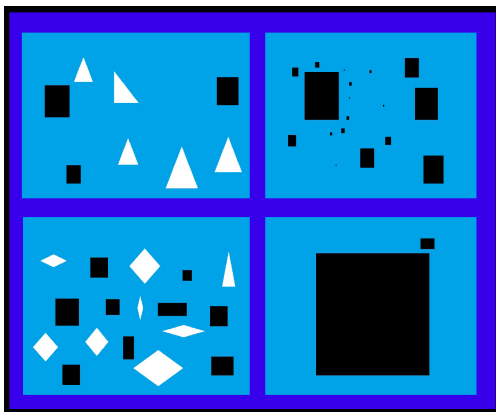


Figure 1: Detección Reconocimiento y Censurado de Cuadrados y Rectángulos

4.2 Detección de Triángulos

En esta clase se crea un método , encargado de extraer todo los contornos del dataset de imágenes, el mismo que será usado más adelante. Primero se lee el dataset, que consta de 100 imágenes, cada imagen consta con figuras básicas como cuadrados, círculos, triángulos y otras figuras adicionales, una vez que se lee las imágenes, se procede a extraer todos los contornos, luego por medio de la extracción de características , en este caso de área y gracias a la función `contourArea()`, la cual calcula el área de un contorno, se procede a dibujar los contornos que solo formen triángulos, es decir aquellos que tengan solo una extensión superior a 600 y que tengan solo tres vértices. Por último, se dibuja el contorno de la imagen detectada en color negro. La detección de triángulos posee ciertas limitaciones, como por ejemplo solo se detectan en ciertas imágenes figuras que no son triángulos, pero si detecta cualquier clase de triángulos como isósceles, equiláteros y rectángulos, los mismos que pueden tener cualquier tamaño. En la figura 2 se puede apreciar los resultados que se obtuvo en las imágenes del dataset.

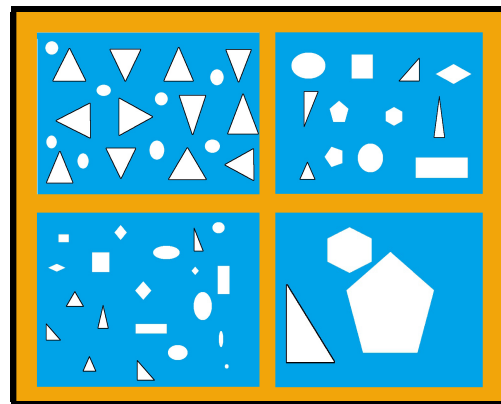


Figure 2: Detección de Triángulos

4.3 Detección de Círculos

En esta clase, primero se lee el dataset, que consta de 100 imágenes, cada imagen consta con figuras básicas como cuadrados, círculos, triángulos y otras figuras adicionales, una vez que se lee las imágenes, se procede a aplicar la Transformada de Hough para círculos, que recibe como parámetros, la imagen, la función del gradiente en los bordes, acumulador de imagen, la mínima distancia, parámetro 1, parámetro 2, el mínimo radio y el máximo radio, luego aplicamos el redondeo, después encontramos los centros y el área. Por último, se dibuja el contorno de la imagen detectada en color negro. La detección de círculos posee ciertas limitaciones, como por ejemplo, no se detectan aquellos círculos demasiado grandes y un poco alargados porque parecen elipses. En la figura 3 se puede apreciar los resultados que se obtuvo en las imágenes del dataset.

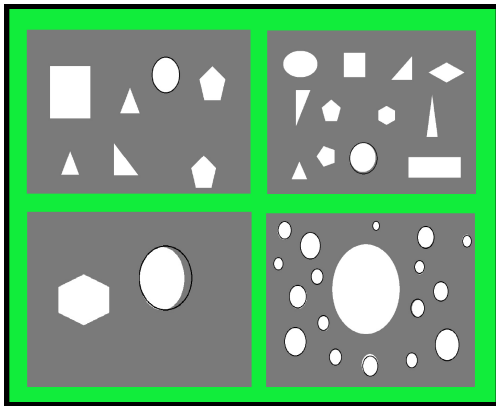


Figure 3: Detección de Círculos

El presente proyecto, cuenta con un repositorio, en donde se puede ingresar mediante este link <https://github.com/Juankrlosx/VisionComputacional.git>, aquí se podrá encontrar todos los programas que se realizaron y una breve explicación sobre su funcionamiento.

5 Análisis de Resultados

En esta sección se presentará los resultados obtenidos y el respectivo análisis que se realizó, para verificar el porcentaje, en el que el sistema detecta la figura geométrica, ya sean los círculos, cuadrados o triángulos, de forma correcta.

En la figura 4 se puede apreciar el análisis en la detección de cuadrados y rectángulos, para su comprensión se realizó una matriz de confusión en donde se presenta de las 100 imágenes del dataset, cuántas acertaron en la detección correcta de los cuadrados y rectángulos, cuántas no detectaron ninguna figura, cuántas detectaron figuras incorrectas y correctas y cuántas no detectaron las figuras a pesar de ser correctas. También se puede observar el porcentaje de precisión del programa y el porcentaje de error, que presenta. El programa de Detección de Cuadrados y Rectángulos tiene una exactitud del 84% y un margen de error del 16%.

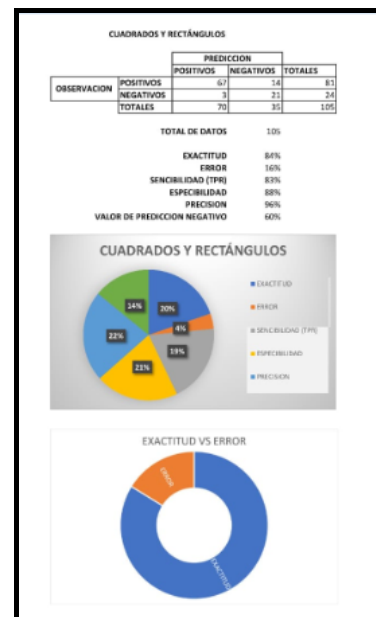


Figure 4: Análisis de Resultados -Detección de Cuadrados y Rectángulos

En la figura 5, también se puede observar el mismo estudio que se realizó anteriormente, pero esta vez se analiza la detección de Triángulos. El programa de Detección de Triángulos tiene una exactitud del 79% y un margen de error del 21%

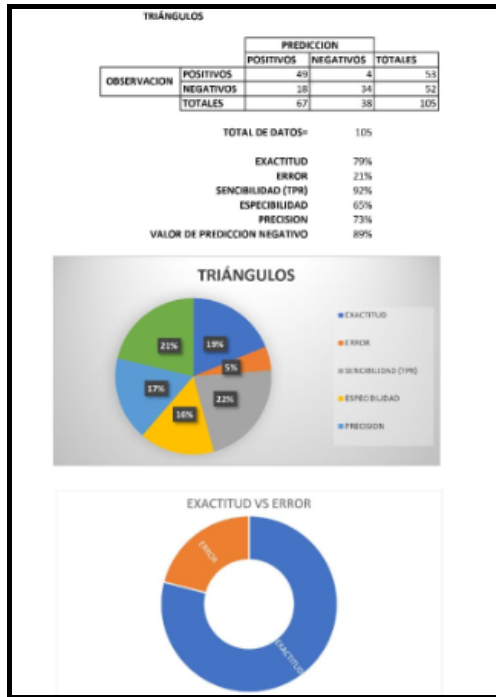


Figure 5: Análisis de Resultados -Detección de Triángulos

En la figura 6, se puede observar el mismo estudio de los resultados obtenidos, pero esta vez se analiza la detección de Círculos. El programa de Detección de Triángulos tiene una exactitud del 62% y un margen de error del 38%

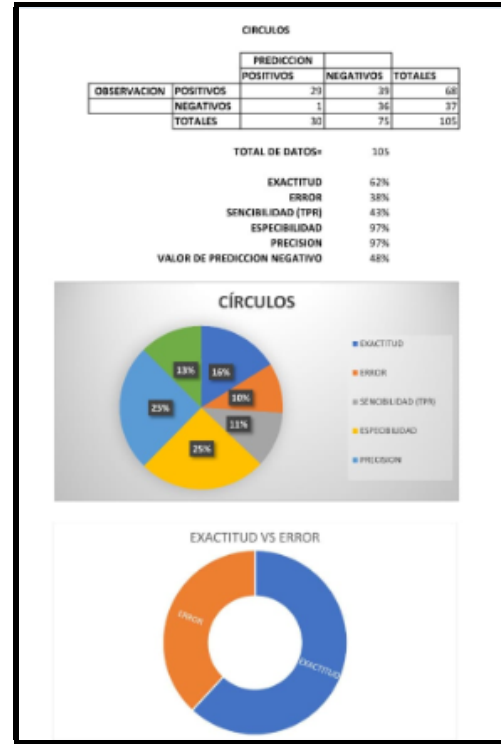


Figure 6: Análisis de Resultados -Detección de Círculos

6 Conclusiones y Trabajos Futuros

6.1 Conclusiones

- El programa de Detección de Triángulos, detecta toda clase de triángulos como isósceles, equiláteros y rectángulos, en sus diferentes tamaños y rotaciones, pero también se puede cambiar el número de vértices para poder detectar figuras de 4,5,6,7,8 lados; por eso el programa tiene una exactitud del 79% y un margen de error del 21%.
- El programa de Detección de Cuadrados y Rectángulos, detecta todos los cuadrados y rectángulos, en diferentes tamaños, que es-



ten correctamente dibujados y que no presenten ningún tipo de rotación; por eso el programa tiene una exactitud del 84% y un margen de error del 16%.

- El programa de Detección de Círculos, detecta todos los círculos, en diferentes tamaños y que estén correctamente dibujados, no detecta aquellos que sean muy grandes y que presenten cierto alargamiento, ya que es considerado como una elipse; por eso el programa tiene una exactitud del 62% y un margen de error del 38%.

- [2] RUIZ Armando, *Reconocimiento de Figuras geométricas mediante Hough*, 2009
- [3] MARTÍNEZ Tania Holly, *La transformación de Hough. Detección de Líneas y Círculos*, 2003
- [4] MANZANARES Feliciano, *Extracción de Características*, 2015
- [5] HOWSE Joseph, PRATEEK Joshi, BEYELER Michael, *OPENCV: Computer Vision Projects With Python*, 2016

6.2 Trabajos Futuros

El presente proyecto tiene muchos campos de investigación, que surgieron en el momento de desarrollar el proyecto y que se espera ser resueltos en el futuro.

A continuación se explicará algunas ideas que surgieron al momento de desarrollar el proyecto.

- 6.2.1** Implementar la detección, no solo de figuras geométricas básicas, como el círculo, el cuadrado, el rectángulo o el triángulo, sino también, tratar que el programa detecte figuras de 8, 9, 10, 11, 12 y 13 lados.
- 6.2.2** Implementar un método en la clase de Cuadrados y Rectángulos, que permita detectar aquellas figuras que poseen alguna rotación.
- 6.2.3** Implementar un método que detecte Círculos, pero sin ocupar la Transformada de Hough.

References

- [1] SOBRADO, Eddie, *Sistema de Visión Artificial para el reconocimiento y manipulación de objetos utilizando un Brazo Robot*, 2003