

# Extracción de características significativas en figuras geométricas primitivas para el reconocimiento de patrones

Carlos Francisco Paz Cuevas

25 de noviembre de 2008

## Resumen

En este trabajo se muestra la implementación de dos técnicas para el reconocimiento de figuras primitivas. Las técnicas utilizadas para la extracción de características son: el cálculo de la curvatura, por medio del cambio de gradiente dentro de la matriz de la imagen en escala de grises y la implementación de una aplicación del Image Processing Toolbox de Matlab en reconocimiento de círculos en imágenes binarias; incorporando la utilización de técnicas de pre procesamiento y el uso de una cámara web.

## 1. Introducción

La toma de decisiones en las industrias automatizadas abarca una diversidad de situaciones o condiciones que puede implicar desde la clasificación de objetos simples hasta la secuencia estructurada que acople las partes de un producto terminado.

Un problema recurrente para clasificar objetos, es la selección de rasgos significativos robustos que describan la naturaleza o forma del objeto. Es mucho mas rápido y fácil de hacerlo para un dispositivo diseñado específicamente para esto, que haber entrenado a un ser humano el cual se cansa y puede cometer errores.

Una forma de resolver el problema es la implementación de algunas técnicas [Web02] para el reconocimiento de figuras geométricas primitivas de las cuales parten las diferentes figuras y formas que se puedan encontrar durante la tarea de reconocimiento.

Este trabajo aborda una solución inicial al problema planteado con anterioridad mediante una aplicación con el Image Processing ToolBox de Matlab para el reconocimiento de círculos. También se aplica el cálculo de la curvatura de una imagen en escala de grises como otra herramienta para la obtención de métricas.

El enfoque principal de la presente investigación, radica en la obtención de métricas representativas de los objetos y los métodos para extraerlas, con la finalidad de conformar un vector de características que ayude en el proceso de clasificación.

## **2. Propósito**

Reconocer figuras geométricas primitivas por medio de la implementación de dos técnicas, para la obtención de características que optimicen su reconocimiento para su clasificación.

## **3. Implementación**

El reconocimiento de las figuras geometricas en nuestro caso, se realiza mediante dos técnicas: una es utilizando las herramientas que nos ofrece el Image Processing Toolbox de Matlab para el reconocimiento de círculos y para el reconocimiento de polígonos (triángulos, cuadrados, rectángulos, etc.) se realizó el cálculo de la curvatura de una imagen mapeada como una matriz con valores de intensidad en escala de grises, para evaluar el cambio de gradiente o el cambio de intensidad de gris para reconocimiento de esquinas.

Cuando se trata de reconocimiento de patrones en imagenes digitales es necesario contar con algun software de proposito especifico, en este caso utilizamos Matlab el cual incorpora diferentes funciones para el procesamiento de imágenes.

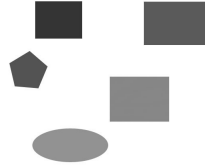


Figura 1: Imagen en Escala de Grises

### 3.1. Pre Procesamiento

Una imagen puede ser analizada como una matriz de pixeles o como una superficie, dependiendo de los valores de intensidad que se generan cuando la imagen es convertida a escala de grises. Esta imagen puede ser obtenida de diferentes fuentes, una cámara digital, una cámara web o puede ser generada por medio de evaluar una función  $f(x)$  y graficarla en un plano, sea cual sea la forma de obtenerla, en este caso analizaremos la imágenes como matrices de pixeles con sus valores de intensidad dentro de la escala de grises.

Una tarea importante es la eliminación del ruido [Dud00] que se haya generado durante la obtención de la imagen, esto se realiza para evitar que en los bordes de los objetos se hayan pixeles con un valor de intensidad muy diferente al de todo el objeto, es decir, tengamos diferentes valores de intensidad para un solo objeto. Existen diferentes métodos de eliminación de ruido, los cuales se dividen en lineales y no lineales, para nuestro caso utilizaremos uno del conjunto de los no lineales, el filtro de la mediana, el cual depende de una ventana que se encuentra alrededor del pixel seleccionado, el tamaño de esta ventana depende de cuantos pixeles vecinos requerimos para el cálculo de la mediana, el valor de intensidad del pixel seleccionado sera modificado de acuerdo a este calculo.

Para nuestro caso, después convertiremos esta imagen de escala de grises en una imagen binaria, esto para diferenciar los objetos del fondo, los objetos deben de contrastar con el fondo, por lo tanto debemos de contar con un fondo homogéneo para un reconocimiento mucho más preciso, ya que si se tiene un fondo heterogéneo, es posible que no se distinga algún objeto o varios objetos puedan ser reconocidos como uno solo. Esta conversión permitira la segmentación de la imagen, donde el fondo tendrá un valor de intensidad uniforme y el valor de intensidad de cada objeto será diferente al del fondo.

### 3.2. Curvatura de la Imagen

Esta técnica se enfoca en calcular en que parte del objeto el gradiente cambia bruscamente de dirección, dicho de otra manera, un cambio brusco en la intensidad de los píxeles, indicándonos en que partes de la imagen, se hayan las esquinas de alguna figura.

Es importante que antes de realizar el cálculo de la curvatura, se realice una división entre los objetos del fondo dentro de la imagen y el fondo esto se logra detectando el borde de cada figura, el fondo se recomienda que sea homogéneo, esto brinda una mejor percepción de los objetos y por lo tanto se logra discriminar de una manera más óptima a cada una de las figuras.

Este proceso se puede realizar con algún método de detección de bordes, existen diferentes métodos que ya se encuentran implementados en Matlab, solo es cuestión de utilizar las funciones dentro del programa, algunos ejemplos de detección de bordes para imágenes son:

- Sobel
- Prewitt
- Roberts
- Laplaciana de Gauss
- Zero Crossing
- Canny

Una vez detectado los bordes, ahora se rellenaran esos objetos para lograr una diferencia logica o binaria con respecto al fondo, la forma en que son rellenados esos objetos, es decir que adquieren una superficie uniforme, es por medio de una estructura que se coloca dentro del objeto y se expande hasta los limites definidos por el objeto, en este caso los bordes previamente detectados.

Para su implementación consideramos lo siguiente:

Sea  $I$  una imagen en escala de grises, la cual haya pasado por una etapa de pre procesamiento y  $I'x$  la primera derivada de la imagen y  $I''x$  la segunda derivada con respecto a  $x$  de la imagen,  $I'y$  y  $I''y$  la primera y segunda derivada con respecto a  $y$ , siendo  $K$  la curvatura de la imagen.

$$K = \frac{IxxIy^2 + IyyIx^2 - 2IxyIxxy}{\sqrt[3]{Ix^2 + Iy^2}} \text{ [AK06]}$$

Ya que tenemos esta matriz con los valores de la curvatura de la imagen, de nuevo discriminamos esos valores, de acuerdo a un umbral que se obtuvo a prueba y error para poder identificar que valores son los que pertenecen a las esquinas de cada objeto. Como se muestra en el cuadro 1, la imagen original y la imagen de la curvatura resultado de la formula anterior aplicada a la imagen original.

Cuadro 1: Imagen Original y Deteccion de Esquinas



### 3.3. Reconocimiento de Circulos con el Image Processing Toolbox de Matlab

Las herramientas que brinda el Image Processing Toolbox de Matlab, permite pre procesar las imagenes y analizarlas, por medio de diferentes funciones que se encuentran organizadas de la siguiente manera:

- Despliegue y Exploración
- Transformación y Registro
- Análisis y Estadísticas
- Aritmética
- Mejoramiento y Restauración
- Filtros
- Operaciones Mofológicas

De estas funciones se toman para el reconocimiento de figuras circulares, estas funciones se adecuan a la necesidad de reconocer estas figuras, por ejemplo, existen funciones de relleno de objetos las cuales tienen un estructura circular, esto ayuda en el proceso a delimitar los objetos llenándolos, y por lo tanto permite diferenciarlos del fondo.

La aplicación consta de casi la misma serie de pasos que se utilizaron para el cálculo de la curvatura. Nuevas funciones se incorporaron a la técnica de reconocimiento de círculos: como la identificación de los objetos por medio de recorrer su frontera o borde, así como también las fronteras que se hayan dentro del objeto, otra función agregada en esta parte es la obtención de ciertas características una vez que se delimitaron los objetos, para esta función es agregar a un vector de características las magnitudes que se van a medir de cada objeto, los cuales están colocadas igualmente en un vector, que se podrá llamar vector de muestras.

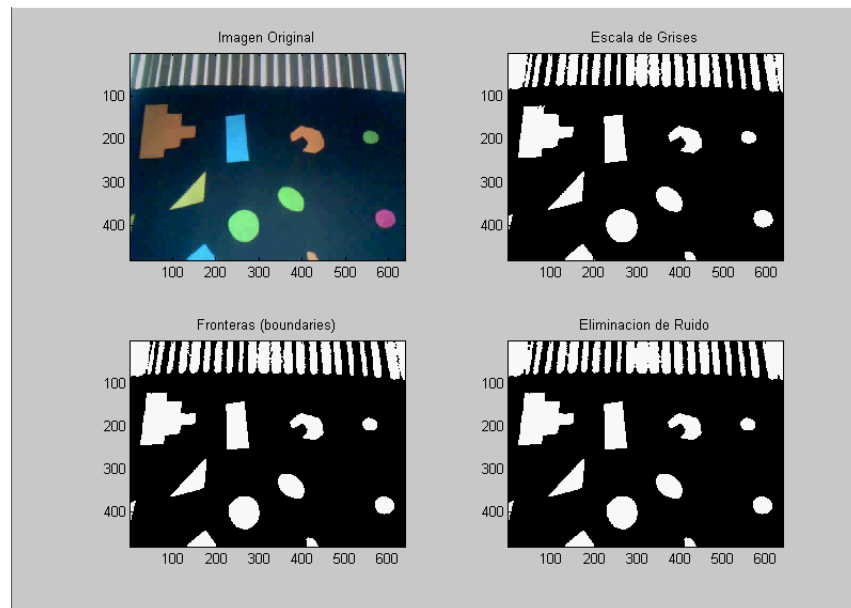


Figura 3: Pre procesamiento en el reconocimiento de círculos

Para reconocer la figura circular se pondrá un interés particular en el perímetro, área y centroide de cada uno de los objetos. estas características permitirán discernir entre los diferentes objetos.

En la implementación de esta parte se realizaron los siguientes pasos:

1. Inicialización de la cámara web (`obj = videoinput('winvideo')`)
2. Captura de la imagen por medio de la cámara web(`ima = getsnapshot(obj);`)
3. Conversión a escala de grises (`rgb2gray`)
4. Eliminación de la variación de píxeles cuando se convierte a blanco y negro (imagen binaria)(`graythresh`)
5. Eliminación de objetos que contengan menos de 30 píxeles (`bwareaopen`)
6. Utilización de la estructura circular para llenar los espacios de los objetos (`strel`)
7. Relleno de objetos (`imfill`)

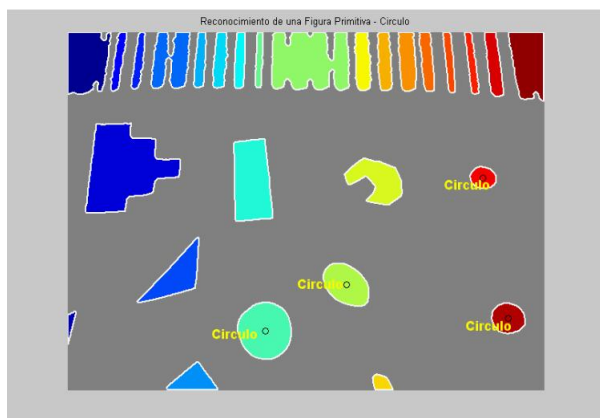


Figura 4: Resultado del reconocimiento de círculos

8. Rastreo de la frontera de cada objeto encontrado (`bwboundaries`)
9. Cálculo de las características más significativas (`regionprops`)

## 4. Trabajos Futuros

Cuando se realizó el programa que calculaba la curvatura de la imagen, se encontraron varias deficiencias en cuanto a la calidad de la imagen, el refinamiento de los bordes de las figuras, los filtros de eliminación de ruido que se requerían e igualmente que figuras son reconocidas.

En cuanto al refinamiento de las líneas es necesario una forma esquelética de los bordes, esto quiere decir hacerlos mucho más finos para que el grosor del borde no exceda demasiados píxeles y por lo tanto esos bordes se consideren como una figura y no el borde de la misma.

Se propone igualmente continuar el trabajo utilizando las imágenes como superficies y obteniendo sus curvas de nivel, se podrá realizar un método en el cual una curva dinámica actúa como una liga que se desprende del borde de la imagen rodeando todas las figuras y delimitándolas.



## 5. Conclusiones

En nuestra experiencia el cálculo de la curvatura se comporto de una manera eficiente al conjuntar los pixeles que conforman a la esquina en una sola región, esto es de gran ayuda cuando se contrasta con la imagen original indicándonos exactamente el cambio de intensidad de la figura sobre sus bordes.

En cuanto a la utilización del Image Processing Toolbox de Matlab, notamos que es una herramienta bastante buena y capaz de realizar diferentes funciones las cuales se adecuaron a nuestra investigación, la implementación de la cámara web como fuente de adquisición de imágenes, el uso de diferentes filtro y conversiones de imágenes, e inclusive ya existen diferentes métodos de detección de bordes, líneas e inclusive esquinas.

Finalmente nuestro vector de características cuenta con las propiedades de las figuras a reconocer, en este caso figuras geométricas, y el cálculo de derivadas y curvatura de la imagen. Nuestro vector cuenta con los siguientes valores:

- Perímetro
- Área
- Centroides (estas tres para el reconocimiento del círculo)
- Primera y segunda derivada de la imagen
- Valor de la Curvatura de la imagen

E igualmente contamos con las siguientes técnicas para el pre procesamiento:

- Conversión a escala de grises
- Eliminación de la variación de pixeles
- Conversión a imágenes binarias
- Eliminación de ruido - Filtro de Mediana
- Detección de Bordes y Fronteras

## Referencias

- [AK06] G. Aubert and P. Kornprobst. *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations (second edition)*, volume 147 of *Applied Mathematical Sciences*. Springer-Verlag, 2006.
- [Dud00] R. Duda. *Pattern Classification*. Wiley, 2000.
- [Web02] A. Webb. *Statistical pattern recognition*. Wiley, 2002.