

14th Argentine Symposium on Technology, AST 2013

## Patrón de Ángulo de Giro para Reconocimiento de Objetos en Imágenes Digitales

Jorge Kamlofsky<sup>1</sup>, María Lorena Bergamini<sup>1</sup><sup>1</sup> CAETI - Universidad Abierta Interamericana

Av. Montes de Oca 725 – Buenos Aires – Argentina

{Jorge.Kamlofsky, Maria.Bergamini}@uai.edu.ar

**Resumen.** Muchas técnicas de visión artificial se basan en reconocer objetos dentro de una imagen mediante la identificación de ciertos patrones en sus bordes. Las fronteras de los objetos en imágenes digitales son susceptibles al ruido generado en el proceso de adquisición de la imagen. En este trabajo se propone un método de aproximación de curvas digitales basado en el concepto de barras digitales, como generalización del concepto de rectas digitales. Se genera un polígono, y el ángulo en cada uno de sus vértices se considera un estimador de la curvatura. Este estimador permite identificar características geométricas del objeto, como ser concavidad, convexidad, puntos angulosos, etc, mediante el análisis de la evolución de dicho parámetro a lo largo de la curva. Así, se obtiene una representación simplificada de la forma que implica una cantidad reducida de puntos en el espacio longitud-curvatura, y es poco sensible al ruido.

**Palabras claves:** Visión Artificial, Formas Digitales, Análisis de Bordes, Identificación de Patrones.

### 1 Introducción

La visión artificial se ocupa de identificar objetos dentro de las imágenes digitales reconociendo en ellos sus patrones característicos. Actualmente existe una gran cantidad de aplicaciones que usan visión artificial basada en topología digital para la eficiente identificación de objetos. Las estrategias usadas para la identificación se basan en análisis de bordes, o análisis de toda la región que comprende el objeto [12].

Un objeto digital es un conjunto conexo de puntos en una grilla digital, y el borde del mismo puede verse como una o varias curvas cerradas simples.

El conjunto de puntos que constituyen el borde de una figura es de gran interés en el análisis de objetos dentro de imágenes, ya que muchas características del objeto (convexidad, dimensiones, agujeros, etc.) pueden estudiarse por medio de su borde.

El proceso de adquisición y digitalización de una imagen introduce ruido. Ese ruido tiene notoria influencia en las fronteras de los objetos, dificultando su estudio, ya que el análisis de bordes por geometría diferencial requiere el análisis de longitud,

rectitud, curvatura y convexidad [2], características que se ven afectadas por las irregularidades del borde.

El reconocimiento de formas puede hacerse por una comparación píxel a píxel entre dos imágenes, o utilizando ciertos atributos basados en características geométricas para medir la similitud entre dos formas. Claramente, la primera alternativa usa más intensivamente los recursos computacionales (memoria y procesador), mientras que la segunda opción aprovecha características distintivas de las formas, para identificarlas con pocos datos, pero sin perder información importante del objeto analizado.

La estimación de características geométricas de un objeto digital es fundamental en el proceso de reconocimiento automático de formas. Algunas de dichas características son perímetro, tangente, circularidad, curvatura, conexidad, partes cóncavas, partes convexas, partes rectas, puntos angulosos, etc. La mayoría de esas características pueden identificarse analizando sólo la curva de borde del objeto.

Una tarea central en geometría digital es el reconocimiento rectas digitales [11]. Varios enfoques del análisis diferencial en geometría digital se basan en reconocimiento de segmentos de líneas rectas [1,7], o estimación de derivadas por convolución [8].

La definición de curvatura para curvas digitales ha dado lugar a diversos estimadores. Considerando a una curva digital como la digitalización de una curva suave continua, los métodos de estimación se basan en derivada de la tangente, el módulo de la segunda derivada de una parametrización, o mediante la inversa del radio del círculo osculante [3]. Por su parte, otros autores proponen una estimación basada en optimización global [5] o en arcos maximales [10]. Varios estimadores se basan en medir segmentos digitales, lo cual es muy sensible a la presencia de ruido en la imagen. En [9] se trata el ruido en imágenes digitales mediante el concepto de segmentos borrosos. Este concepto se fundamenta en la noción de barra digital [6], que es una generalización de la definición de segmentos digitales.

En [4] se propuso un método de aproximación de curvas por polígonos basado en la noción de ancho de la cáscara convexa de un conjunto de puntos.

El presente trabajo propone un método de representación de formas a fin de ser aplicado para el reconocimiento confiable de objetos, en forma eficiente. El método se basa en aproximar inicialmente el borde de un objeto por un polígono y obtener luego una representación de la evolución del ángulo de giro de dicho borde, como estimación discreta de la curvatura. La poligonalización de curvas presentado en [4] requiere la obtención y medición de la cáscara convexa de un conjunto de puntos que se va modificando con la adición de puntos en cada iteración, a fin de detectar aproximaciones a segmentos. En la presente propuesta, la detección de segmentos se lleva a cabo con un procedimiento basado en barras digitales, que implica menor costo computacional. La aproximación se hace teniendo en cuenta una tolerancia del error cometido al reemplazar puntos del borde por segmentos rectos. Como consecuencia de la aproximación, el borde se suaviza, eliminándose la incidencia del ruido en la imagen digital, además de reducir notablemente la cantidad de puntos de la curva sin perder información relevante.

La representación de la evolución del giro puede verse como la versión discreta de la evolución de la curvatura en figuras continuas planas. La idea del presente trabajo

no es estimar la curvatura en cada punto de la curva digital, si no visualizar la evolución global de dicho parámetro a lo largo de la curva. La representación de la evolución del giro es invariante frente a movimientos rígidos (traslación, rotación, simetrías) y también ante escalamiento uniforme, y permite identificar características geométricas de la forma analizada (concavidad, convexidad, bordes rectos, etc.).

En la siguiente sección se detalla el método propuesto para la poligonalización de curvas y la obtención del patrón de giro. En la sección 3 se muestran resultados obtenidos al aplicar el procedimiento a un conjunto de imágenes. Finalmente, el trabajo concluye describiendo el uso del patrón de giro para identificación de formas, que constituye la aplicación a evaluar, como continuación del presente trabajo.

## 2 Método propuesto

### 2.1 Poligonalización de curvas digitales

La curva de borde es un conjunto discreto de puntos, y en general, no es suave, no sólo por su naturaleza discreta, sino también por la presencia de ruido en la imagen. A fin de suavizar y simplificar la curva de borde, se propone un algoritmo para aproximar una curva digital por un polígono. Para ello, deben detectarse arcos de la curva dada que se puedan aproximar por segmentos de recta. En este trabajo no se pretende reconocer exactamente los conjuntos de puntos provenientes de digitalización de rectas, sino identificar conjuntos de puntos que están aproximadamente alineados. Esta aproximación se hace teniendo en cuenta una tolerancia de aproximación.

En geometría digital se definen rectas digitales a partir del concepto de barra digital. Una barra digital de pendiente  $a/b$ , límite inferior  $c$  y ancho  $d$  (con  $a, b, c, d$  enteros,  $d > 0$  y  $\text{mcd}(a,b)=1$ ) es el conjunto

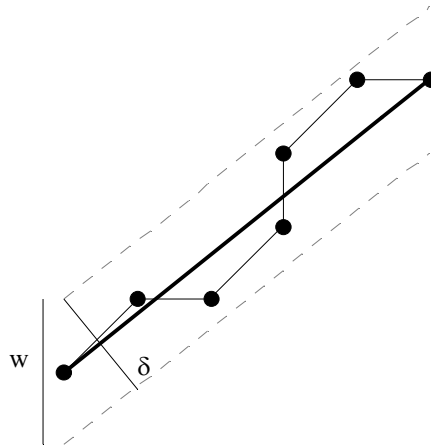
$$D_{a,b,c,d} = \{(x, y) \in Z^2 : c \leq ax - by < c + d\} \quad (1)$$

Este conjunto está formado por puntos de coordenadas enteras entre las rectas paralelas  $y = (a/b)x - (c/b)$  y  $y = (a/b)x - (c + d - 1)/b$ . La distancia vertical entre estas rectas es  $w = (d - 1)/|b|$ , y la distancia perpendicular es  $\delta = aw/\sqrt{a^2 + b^2}$ . Está demostrado que  $D_{a,b,c,\max\{|a|,|b|\}}$  es un conjunto de puntos grilla que es digitalización de una recta de pendiente racional [6].

En este trabajo se consideran barras digitales de ancho perpendicular  $\delta=2\varepsilon$ , siendo  $\varepsilon$  una tolerancia prefijada. Dado un arco de la curva de borde de un objeto, asumimos que se puede aproximar por un segmento de recta si todos sus puntos están dentro de una barra de ancho perpendicular  $2\varepsilon$ . La figura 1 muestra un conjunto de siete puntos de un arco de curva digital, dentro de una barra digital de ancho perpendicular  $\delta$ .

Dada la lista de puntos de borde  $\{p_1, p_2, \dots, p_m\}$ , el algoritmo propuesto divide la lista en sublistas maximales de puntos consecutivos  $\{p_s, p_{s+1}, \dots, p_{s+j}\}$ , que se sustituyen por un lado del polígono. Los puntos en cada sublista se consideran aproximadamente alineados, al verse comprendidos en una barra digital de ancho perpendicular

$2\varepsilon$ , con eje en el segmento de recta continuo que une  $p_s$  con  $p_{s+j}$ . Se dice que la sublista es maximal, porque al incorporar el siguiente punto de la curva,  $p_{s+j+1}$ , no se verifican esas condiciones. Así,  $p_s$  y  $p_{s+j}$  son vértices del polígono aproximante.



**Fig. 1.** Barra digital de ancho vertical  $w$  y ancho perpendicular  $\delta$ .

El algoritmo construye sucesivamente los segmentos consecutivos del polígono. Para cada punto de la curva, se analiza si éste puede ser admitido en el lado que se está construyendo, o debe pertenecer a un nuevo lado.

El procedimiento para determinar la admisión del punto al lado recibe como argumentos el lado que se está construyendo (la lista de puntos ya admitidos en él), y el nuevo punto candidato a pertenecer al mismo lado. Siendo  $p_s, p_{s+1}, p_{s+2}, \dots, p_{s+r}$  los puntos ya admitidos en el lado y  $p_{s+r+1}$  el nuevo candidato, el procedimiento mide la distancia de  $p_{s+1}, p_{s+2}, \dots, p_{s+r}$  a la recta que pasa por  $p_s$  y  $p_{s+r+1}$ . Si todas estas distancias son menores que la tolerancia  $\varepsilon$  especificada, entonces todos los puntos entre  $p_s$  y  $p_{s+r+1}$  están dentro de la barra digital de eje en el segmento con extremos  $p_s$  y  $p_{s+r+1}$ , de ancho  $2\varepsilon$ . En este caso, el nuevo punto queda admitido en el lado. En caso contrario, se comienza un nuevo lado, con vértice inicial en  $p_{s+r}$ .

El algoritmo da como salida el conjunto de extremos iniciales de los lados, que son los vértices del polígono aproximante.

Claramente, el método propuesto depende del punto inicial de la curva de borde, pero esa dependencia no hace perder características globales fundamentales de la geometría del objeto. También depende de la tolerancia especificada. Imponiendo tolerancias grandes, se generan lados que abarcan muchos puntos y polígonos de pocos vértices. Así, podrían perderse detalles de menor escala, aunque también hace al método menos sensible al ruido. Con tolerancias más pequeñas se obtienen lados de pocos puntos, y por lo tanto, polígonos de más lados. En el caso límite de tolerancia  $\varepsilon = 0$ , el proceso de poligonalización sólo elimina los puntos intermedios de segmentos de rectas digitales de pendiente múltiplo de  $\pi/4$ .

## 2.2 Obtención del patrón de giro

Dado el polígono obtenido como aproximación de una curva digital, se proponen una representación del mismo basado en el patrón de giro.

Sea  $C$  una curva regular, parametrizada en el plano  $x$ - $y$  con  $\alpha(s) = (x(s), y(s))$ , donde  $s$  es el parámetro longitud de arco,  $s \in [0, L]$ , con  $L =$  longitud de la curva.

La curvatura en el punto  $\alpha(s)$  es  $\kappa(s) = |d\mathbf{T}/ds|$ , siendo  $\mathbf{T}(s) = \alpha'(s)/|\alpha'(s)|$  el vector unitario tangente a la curva en el punto  $\alpha(s)$ . La curvatura también puede pensarse como el cambio instantáneo de la inclinación del vector tangente. Es decir,  $\kappa(s) = \lim_{h \rightarrow 0} (\theta(s+h) - \theta(s))/h$ , con  $\theta =$  dirección de  $\mathbf{T}$ .

Sea  $l(s)$  la longitud de la curva desde  $\alpha(0)$  hasta  $\alpha(s)$  y  $\kappa_{acum}(s) = \int_0^s \kappa(r) dr$ . Obviamente,  $\kappa_{acum}(L) = 2\pi$  para curvas cerradas simples.

Sea  $\beta(s)$  la curva en el plano  $l$ - $\kappa$ , que representa la curvatura acumulada  $\kappa_{acum}(s)$  en función de la longitud normalizada, es decir  $\beta(s) = (\lambda(s), \kappa_{acum}(s))$ , con  $\lambda(s) = l(s)/L$ , con inicio en  $(0,0)$  y final en  $(1, 2\pi)$ . Esta curva también puede describirse como la gráfica de la función  $\kappa_{acum}(\lambda)$ , con  $\lambda \in [0, 1]$ .

Extendamos estas definiciones para el caso en que  $C$  sea un polígono de vértices  $v_1, v_2, \dots, v_n$ . En lugar de parametrizar con un parámetro continuo, la curva se describe por sus vértices. La curva poligonal se representa con  $\alpha(i) = (x(i), y(i))$ , para  $i=1, 2, \dots, n$  (ordenados tal que el polígono se recorre en sentido horario). La curvatura  $\kappa(i)$  en el vértice  $v_i$  es el ángulo orientado formado por los segmentos orientados  $v_{i-1}v_i$  y  $v_iv_{i+1}$  (tomando los índices módulo  $n$ ), es decir, el ángulo de giro del polígono en el vértice  $v_i$ . En este trabajo se considera positivo un giro en sentido horario. Sea  $l(i)$  la suma de las longitudes de los segmentos del polígono desde  $v_1v_2$  hasta  $v_iv_{i+1}$  (módulo  $n$ ) y  $L$  la longitud total del polígono. Finalmente, definimos

$$\kappa_{acum}(i) = \sum_{j=1}^i \kappa(j) \quad (2)$$

es decir, el ángulo de giro acumulado desde el primer vértice hasta el vértice  $i$ . Claramente,  $\kappa_{acum}(n) = 2\pi$ .

La evolución de giro de un polígono es la curva lineal por tramos en el plano  $l$ - $\kappa$ , cuyos vértices son  $\beta(i) = (\lambda(i), \kappa_{acum}(i))$ , siendo  $\lambda(i) = l(i)/L$  la longitud normalizada, con  $i = 0, 1, 2, \dots, n$ . El punto inicial de esta curva se toma como  $\beta(0) = (0,0)$ . También puede pensarse esta curva como la gráfica lineal por tramos de la función  $\kappa_{acum}(\lambda)$  con  $\lambda \in [0, 1]$ , donde ocurre que  $\kappa_{acum}(0) = 0$  y  $\kappa_{acum}(1) = 2\pi$ .

Esta función identifica la forma del polígono, independientemente de la posición y el tamaño del mismo en el plano. Es decir, es invariante frente a traslaciones, rotaciones y escala.

Además, brinda información sobre las características geométricas de la curva. Los intervalos de  $\lambda$  donde  $\kappa_{acum}(\lambda)$  es creciente, determinan las partes convexas del

polígono. Y viceversa, los intervalos donde  $\kappa_{acum}(\lambda)$  es decreciente determinan las partes cóncavas (para el borde exterior de un objeto; al revés en el caso de curvas interiores, bordes de agujeros, recorridas también en sentido horario). De esta forma, los máximos y mínimos relativos de esta función dan los puntos de inflexión.

### 2.3 Dependencia del punto inicial

El punto inicial de la curva analizada, y por lo tanto, del polígono obtenido al aproximarla, depende del algoritmo usado para detectar bordes de objetos en imágenes binarias, y en general, depende de la orientación del objeto. Es decir, aún usando el mismo algoritmo de detección de bordes, el punto inicial depende de la rotación del objeto.

En particular, el algoritmo para detectar bordes aplicado en este trabajo comienza a recorrerlo en el punto más arriba y más a la izquierda de la figura. Claramente, este punto depende de la rotación de la imagen.

La curva  $\kappa_{acum}(\lambda)$  puede extenderse en forma continua periódica al intervalo  $[1,2]$ , haciendo  $\kappa_{acum}(\lambda) = \kappa_{acum}(\lambda - 1) + 2\pi$ , si  $\lambda \in [1,2]$ .

De esta forma, si el recorrido de los vértices del polígono empieza en el vértice  $k$ , es decir, se recorren en el orden  $v_k, v_{k+1}, \dots, v_n, v_1, v_2, \dots, v_{k-1}$ , la función obtenida como representación de ángulo de giro es  $\tilde{\kappa}_{acum}(\lambda) = \kappa_{acum}(\lambda + \lambda_0) - \kappa_{acum}(\lambda)$ , donde  $\lambda_0 = l(k) / L$ , y tomando  $\kappa_{acum}$  extendida como se explicó antes.

Para normalizar la representación, y que sea independiente del vértice inicial, se considerará la representación uniformizada  $\tilde{\kappa}_{acum}$  tomando como inicio el vértice  $k$  tal que  $\kappa(k)$  sea el de mayor valor absoluto.

### 2.4 Características de la representación con el patrón de giro.

Es sabido que las curvas continuas planas son identificadas con la función curvatura o con  $\kappa_{acum}(s)$ . Sin embargo, en curvas digitales, la naturaleza discreta impide que esta propiedad se mantenga con precisión.

Llamamos distancia entre dos patrones  $\kappa_1$  y  $\kappa_2$  al valor

$$D(\kappa_1, \kappa_2) = \int_0^1 (\kappa_1(\lambda) - \kappa_2(\lambda))^2 d\lambda \quad (3)$$

La idea de que la representación del patrón de giro identifica una curva digital puede expresarse en los siguientes términos. Dos representaciones  $\kappa_1$  y  $\kappa_2$  de la misma forma, obtenidas de imágenes distintas y posiblemente a distinta escala, son tales que la distancia entre ellas es chica. Y viceversa, si dos representaciones son cercanas, las curvas que las originan son similares.

Junto con la representación del patrón de giro, pueden abstraerse características geométricas de la forma analizada, a fin de realizar un primer filtrado entre las formas, como se describe a continuación.

Dada una forma  $S$  y su patrón de giro  $\kappa_{acum}(\lambda)$ , se define la zona de concavidad  $Cv_S =$  conjunto de intervalos de  $\lambda$  en los que  $\kappa_{acum}(\lambda)$  es decreciente, y la zona de convexidad  $Cx_S =$  conjuntos de intervalos de  $\lambda$  en los que  $\kappa_{acum}(\lambda)$  es creciente. Además, se define la zona significativa recta,  $L_S =$  conjunto de intervalos maximales de  $\lambda$  donde  $\kappa_{acum}(\lambda)$  es lineal, cuya longitud es mayor que 0.15. Esta condición sobre la longitud de cada intervalo se impone para que este conjunto refleje efectivamente las partes rectas de la forma analizada, y no sean producto de la aproximación de partes curvas.

### 3 Aplicación del método propuesto

El procedimiento se aplicó a una serie de imágenes binarias, con distintas tolerancias. La figura 2 muestra tres aproximaciones de tres formas, para ejemplificar los resultados (imágenes de letra m, letra c y letra f), usando tolerancias 0, 1 y 4.

Es claro que sobre las partes del borde que son rectas, hay pocos puntos del polígono, mientras que en las partes del borde que no son rectas, los vértices de la aproximación se acumulan. Puede verse también que usando tolerancia 4, la curva se deforma considerablemente.

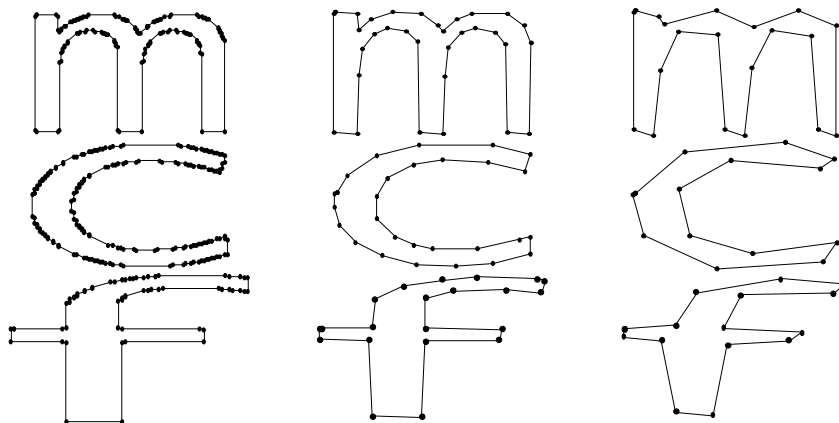


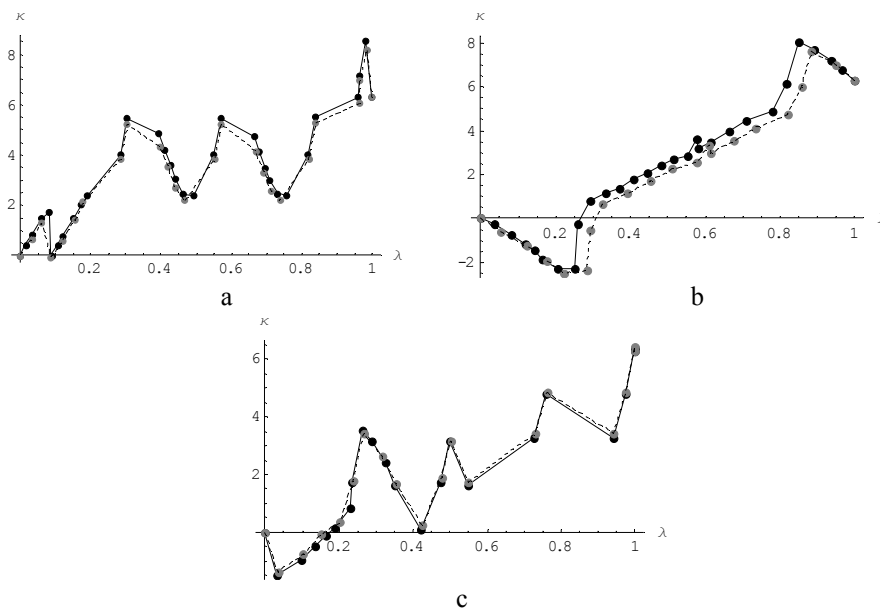
Fig. 2. Aproximación poligonal de objetos con tolerancias 0, 1 y 4

Asimismo, la tabla 1 informa la cantidad de puntos de borde de distintos objetos, y la cantidad de puntos resultantes en el polígono aproximante con las distintas tolerancias. En la tercera columna se da en cada caso, junto con la cantidad de puntos, el porcentaje de puntos que resultan usando tolerancia 0. Este valor da una idea de la proporción de partes rectas y partes no rectas de cada forma. Por ejemplo, la frontera de la imagen de la letra s no tiene muchas partes rectas, y la eliminación de puntos por poligonalización con tolerancia 0 es apenas 48%. La letra h es la forma (entre las incluidas en la tabla) que tiene mayor proporción de partes rectas, y la aproximación con tolerancia 0 elimina 88% de los puntos del borde. La experimentación con distin-

tas formas y distintas tolerancias mostró que, en general, tolerancias más grandes hacen perder detalles de la forma, aunque sin una notable ganancia en reducción de puntos. Puede verse en la tabla 1 que la cantidad de puntos del polígono obtenido con tolerancia 4 no es mucho menor que la cantidad de puntos en el polígono obtenido con tolerancia 2.

**Tabla 1.** Puntos resultantes en las aproximaciones

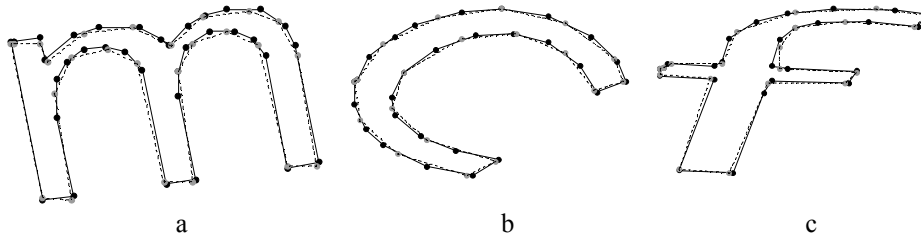
Objeto	Puntos borde	Cantidad de puntos en polígonos			
		$\varepsilon=0$	$\varepsilon=1$	$\varepsilon=2$	$\varepsilon=4$
Letra m	640	126 (19%)	33	25	21
Letra c	368	182 (49%)	19	21	15
Letra f	423	61 (14%)	22	19	17
Letra d	452	110 (24%)	22	16	13
Letra s	429	224 (52%)	39	27	18
Letra h	549	71 (12%)	20	16	14
Forma 1	556	213 (38%)	38	27	16
Forma 2	1051	482 (46%)	90	58	41



**Fig. 3.** Patrones de giro obtenidos con poligonalización con tolerancia 1 (línea llena, puntos negros) y tolerancia 2 (línea de puntos, puntos grises). a. Letra m. b. Letra c. c. Letra f.

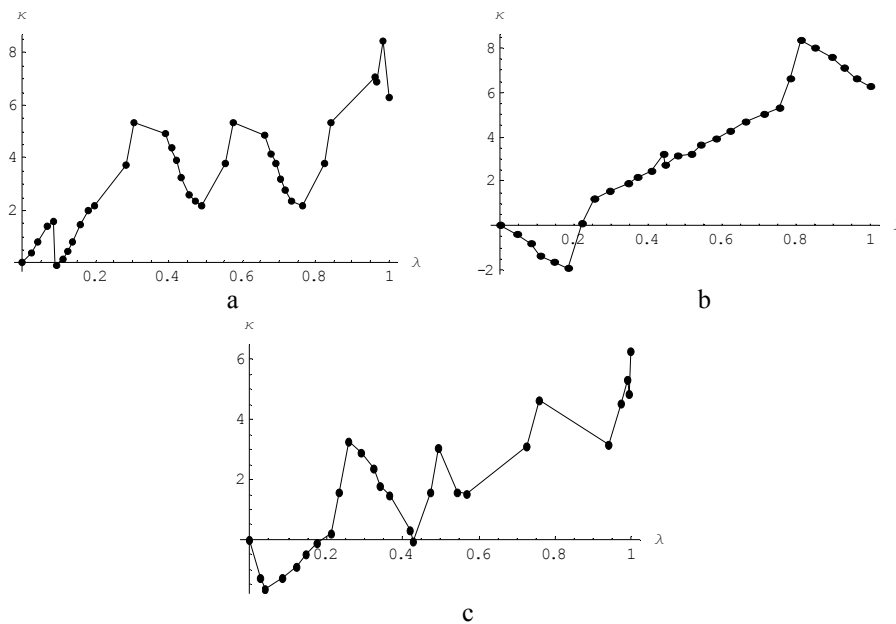
En la figura 3 se muestra la representación del patrón de giro  $\kappa_{acum}(\lambda)$  para tres objetos digitales, representando la letra m, la letra c y la letra f. Puede notarse que, a pesar de ciertas diferencias locales, la evolución global del ángulo de giro obtenido con diferentes tolerancias es similar para cada forma.





**Fig. 4.** Poligonización de imágenes rotadas, con tolerancia 1(en línea llena y vértices negros) y tolerancia 2(en línea de puntos, vértices grises). a - Letra m rotada 10°. b - Letra c rotada -45°. c - Letra f rotada -15°

Como se explicó en la sección 2, el patrón permite obtener información geométrica de la forma analizada. El patrón de la letra m indica que dicha forma tiene 4 partes convexas, que se desarrollan a partir del punto inicial, en los intervalos de longitud  $[0,0.08]$ ,  $[0.09,0.3]$ ,  $[0.48,0.56]$  y  $[0.75,0.98]$ . Y esta partición en partes cóncavas y convexas es una característica propia de esta figura.



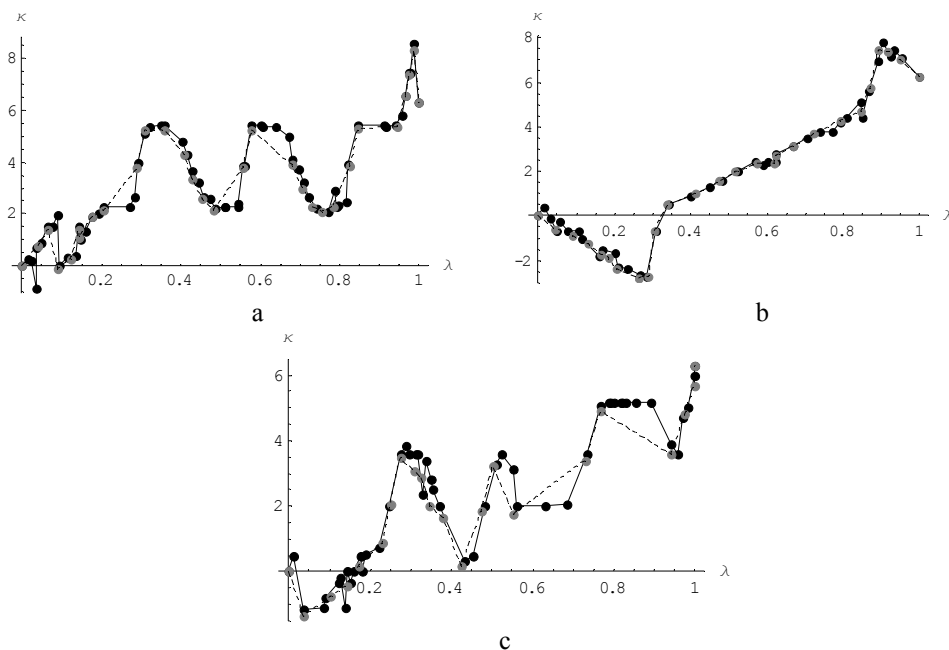
**Fig. 5.** Representación con patrón de giro de objetos rotados obtenidos a partir de la poligonización con tolerancia 1. a - Letra m. b - Letra c. c - Letra f.

Se consideraron también imágenes rotadas de los objetos analizados. La figura 4 muestra el resultado de la poligonización de estas figuras con tolerancia 1 y tolerancia 2. Para estos objetos, el algoritmo de obtención de curvas de borde construye una curva con inicio en un punto distinto al caso de la imagen sin rotar. Por lo tanto, el polígono aproximante, también tiene vértice inicial en un punto distinto.

Para que la representación del patrón de giro sea comparable con el caso anterior, la representación fue uniformizada, tal que se inicie en el vértice de mayor ángulo (en valor absoluto). La figura 5 muestra el patrón de giro obtenido a partir de la curva de borde de la letra m, la letra c y la letra f, rotadas. Se observa una aceptable conformidad con el patrón obtenido en la figura 3. Las mismas características geométricas de las formas analizadas pueden obtenerse a partir de estos patrones.

La letra c está formada por una parte cóncava y una parte convexa (correspondientes a las longitudes en los intervalos  $[0,0.23] \cup [0.82,1]$  y  $[0.23, 0.82]$  respectivamente). Un análisis similar puede hacerse con la representación en la figura 3.c.

Por otra parte, cada segmento de esta función lineal por tramos corresponde a un lado del polígono, y la longitud del segmento coincide con la longitud normalizada del lado que representa. Así, también puede extraerse información sobre las partes rectas de la forma analizada. Por ejemplo, las partes rectas más extensas de la forma de la letra f, de acuerdo con la figura 3.c se extiende en los intervalos  $[0.55, 0.73]$  y  $[0.76, 0.94]$ . La imagen de la letra c no tiene partes rectas significativas.



**Fig. 6.** Patrones de giro en imágenes con ruido, usando tolerancia 1 (línea llena, puntos negros) y tolerancia 2 (línea de puntos, puntos grises). a. Letra m. b. Letra c. c. Letra f.

Si la imagen está distorsionada por ruido, los bordes suelen no estar bien definidos. A fin de mostrar el comportamiento del algoritmo propuesto en este trabajo, se incorporó ruido aleatorio a las imágenes analizadas. Dado que el proceso de poligonalización se basa en barras digitales, un adecuado ancho de las mismas puede absorber el ruido. Existe un compromiso entre el ancho de las barras y la profundidad del ruido.

En la figura 6 se expone el patrón de giro de los tres objetos, con tolerancias 1 y 2. Con tolerancias hasta  $\varepsilon = 4$ , el patrón es casi idéntico al caso de tolerancia 2.

Como es esperable, usando tolerancia  $\varepsilon = 1$ , el patrón difiere bastante del obtenido de la imagen sin ruido. Esto es porque el ancho de las barras digitales es igual al ancho de un pixel, y el ruido tiene incidencia protagónica en el proceso de poligonalización. Sin embargo, con  $\varepsilon = 2$ , el patrón obtenido se asemeja al de la figura sin ruido.

#### 4 Método de reconocimiento de formas usando patrón de giro

Dada una forma  $S$  que se quiere reconocer, se compara con un conjunto de formas conocidas  $S_j$ . Sea  $\kappa_{acum}(\lambda)$  el patrón de giro de  $S$ . Sean  $\alpha_j(\lambda)$  patrones de las formas  $S_j$ , con  $j = 1, 2, \dots, q$ , candidatas a empatarse con  $S$ . Entonces,  $S$  se identifica con  $S_k$  cuyo patrón se encuentra a menor distancia de  $\kappa_{acum}(\lambda)$ , entre todas las candidatas. Es decir,

$$D(\kappa_{acum}, \alpha_k) \leq D(\kappa_{acum}, \alpha_j) \quad \forall j = 1, 2, \dots, q \quad (4)$$

Las formas candidatas a empatarse con  $S$  se obtienen por un filtrado inicial que tiene en cuenta la partición del borde en zonas cóncavas y convexas, zonas rectas.

Específicamente, siendo  $Cv_{S_j}$ ,  $Cx_{S_j}$  y  $L_{S_j}$  las zonas cóncavas, zonas convexas y zonas significativas rectas de la forma  $S_j$ , ésta pasa el filtrado si estos conjuntos son similares a los correspondientes a la forma analizada,  $S$ . Formalmente, se calcula

$$\begin{aligned} Cv(S, S_j) &= Cv_S \cap Cv_{S_j} \\ Cx(S, S_j) &= Cx_S \cap Cx_{S_j} \\ L(S, S_j) &= L_S \cap L_{S_j} \end{aligned} \quad (5)$$

El filtrado es positivo si se verifican las siguientes desigualdades.

$$\begin{aligned} |Cv(S, S_j)| &\leq \mu |Cv_S| \\ |Cx(S, S_j)| &\leq \mu |Cx_S| \\ |L(S, S_j)| &\leq \mu |L_S| \end{aligned} \quad (6)$$

donde  $|\cdot|$  denota la longitud del conjunto, y  $\mu$  es una tolerancia de similaridad. Así, luego del filtrado, subsisten las formas que tienen cierta similaridad con  $S$ .

#### 5 Conclusiones

La poligonalización es un procedimiento que permite representar una curva digital con una considerable reducción de puntos, significando menor requerimiento de espa-

cio de almacenamiento, sin perder información relevante.

El algoritmo de obtención del polígono es de baja complejidad computacional, requiere sólo el cálculo de distancias para cada punto. Esto constituye una ventaja frente a otros procedimientos que implican el cálculo de la cáscara convexa y su ancho. Además, el método es poco sensible al ruido, usando una adecuada tolerancia.

La representación de formas mediante el patrón de giro, como una extensión del concepto de curvatura acumulada, implica la identificación de un conjunto de puntos en el espacio  $\lambda$ - $\kappa$ . Esto también supone una cantidad reducida de memoria requerida.

El patrón de giro permite identificar formas, y por lo tanto, constituye una herramienta útil en un proceso de reconocimiento de objetos en una imagen digital. Las potencialidades del método de reconocimiento se estudiarán en trabajos futuros.

## Referencias

1. Debled-Rennesson I., Reveillès J.-P.: A linear algorithm for segmentation of digital curves. *Int. J. Patt. Recogn. Artif. Intell.* 9 (4), 635 – 662 (1995).
2. Eckhardt U.: Digital Lines and Digital Convexity. In: Bertrand G. Imiya A. Klette R.(eds.) *Digital and Image Geometry*. LNCS, vol. 2243, pp. 209-228. Springer, Heidelberg (2001).
3. Hermann S., Klette R.: A comparative study on 2D curvature estimators. *Int. Conference on Computing: Theory and Applications (ICCTA'07)*, pp. 584-589 (2007).
4. Kamlofsky J., Bergamini, M.: Aproximación de Formas de Objetos Digitales por Polígonos. VII Jornadas Argentinas de Robótica (2012).
5. Kerautret B., Lachaud J.-O.: Curvature Estimation along Noisy Digital Contours by Approximate Global Optimization. *Patt. Recogn.* 42(10), 2265 – 2278 (2008).
6. Kettle R., Rosenfeld A.: Digital Straightness – a review. *Discrete Appl. Math.* 139, 197-230 (2004).
7. Lachaud J-O, Vialard A., Vieilleville F.: Fast, accurate and convergent tangent estimation on digital contours. *Image Vision Comput.* 25(10), 1572–1587 (2007).
8. Malgouyres R., Brunet F., Fourey S.: Binomial convolutions and derivatives estimation from noisy discretizations. In: Coeurjolly D., Sivignon I., Tougne L., Dupont F. (eds.) *Discrete Geometry for Computer Imagery*, LNCS, vol. 4992, pp. 370-379. Springer, Heidelberg (2008).
9. Nguyen T.P., Debled-Rennesson I.: Curvature Estimation in Noisy Curves. In: Kropatsch W., Kampel M., Hanbury A. (eds.) *Computer Analysis of Images and Patterns*, LNCS, vol. 4673, pp 474-481, Springer, Heidelberg (2007).
10. Roussillon T., Lachaud J-O.: Accurate Curvature Estimation Along Digital Contours With Maximal Digital Circular Arcs. In: Aggarwal J., Barneva R., Brimkov V. Koroutchev K., Korutcheva E. (eds.) *Combinatorial Image Analysis*. LNCS, vol 6636, pp 43-55, Springer, Heidelberg (2011).
11. Ucska-Wehlou H. Digital lines, Sturmian words and continued fraction. *Uppsala Dissertations in Mathematics* 65 (2009).
12. Zhang D., Lu G.: Review of shape representation and description techniques, *Patt. Recogn.*, 37 (1),1-19 (2004).