

Patrón de Evolución Discreta de Curvatura y Concavidad para Reconocimiento de Formas

Kamlofsky, Jorge Alejandro - Bergamini, María Lorena

CAETI - Universidad Abierta Interamericana, Facultad de Tecnología Informática

Abstract.

Muchas técnicas de Visión Artificial se basan en el reconocimiento de objetos dentro de las imágenes digitales identificando ciertos patrones. En este trabajo se presentan avances en las investigaciones que pretenden obtener algoritmos para realizar el seguimiento de un objeto en movimiento. Se proponen algoritmos de simplificación de curvas de borde mediante transformación de dichas curvas a polígonos: se logran aproximaciones adecuadas de las curvas y además se permite ignorar efectos de ruido causados por la digitalización. En los vértices de los polígonos se obtienen estimadores de curvatura y concavidad. Su evolución a lo largo de la curva permite identificar características geométricas del objeto, como ser: concavidad, convexidad, tramos rectos, puntos angulosos, entre otros. Se obtiene una representación simplificada de la forma que implica una cantidad reducida de puntos en el espacio longitud-curvatura la cual puede usarse tanto en el proceso de identificación de formas en una imagen como en el proceso de representación de la mismas.

Palabras Clave

Visión artificial, Identificación de patrones, Patrón de evolución del ángulo de giro, Poligonalización de formas, Aproximación de formas digitales por polígonos, Topología digital, Reconocimiento de Objetos, Curvas Digitales, Análisis de Bordes.

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 borde conforma la frontera entre el objeto digital y el fondo de la imagen. Entonces, tener el borde del objeto permite separarlo de la imagen. Un algoritmo simple para la obtención del borde de un objeto digital fue presentado por A. Rozenfeld en [13].

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 en el 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 los recursos computacionales (memoria y procesador) en forma intensiva, 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 de 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 de la estimación de curvatura y concavidad del borde rectificado a lo largo de la curva. Dicha representación simplificada (o curva característica de evolución de curvatura o patrón de giro) permite realizar

comparaciones entre curvas digitales simples, de modo de poder obtener una métrica de similitud entre formas obtenida a partir del cálculo de la distancia entre estas curvas características o patrón de giro.

La poligonalización de curvas presentada 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 formas mediante el patrón de evolución de curvatura 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.).

Mediante la detección de la orientación de la forma de la curva plana es posible obtener patrones de giro uniformes independientemente del punto de inicio del algoritmo.

En la siguiente sección se detallan los dos métodos propuestos para la poligonalización de curvas. En la sección 3 se presenta un estimador de curvatura,

concauidad y la obtención del patrón de giro. En la sección 4 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. La poligonalización de curvas digitales planas

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.

2.1. Poligonalización mediante el uso del concepto de la barra digital

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 $mcd(a,b)=1$) es el conjunto

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

Este conjunto está formado por puntos de coordenadas enteras entre las rectas

$$y = \frac{a}{b}x - \frac{c}{b} \quad \text{y}$$

$$y = \frac{a}{b}x - \frac{c+d-1}{b}. \quad \text{La distancia vertical}$$

entre estas rectas es $w = \frac{d-1}{|b|}$, y la

distancia perpendicular es $\delta = \frac{aw}{\sqrt{a^2 + b^2}}$.

Reveillès [15] demuestra que $D_{a,b,c,\max\{|a|,|b|\}}$ coincide con un conjunto de puntos grilla asignado a una recta digital de pendiente racional, y a la inversa, para cualquier recta digital racional, existen parámetros a, b, c tal que el conjunto de puntos grilla correspondiente a un segmento recto digital coincide con $D_{a,b,c,\max\{|a|,|b|\}}$.

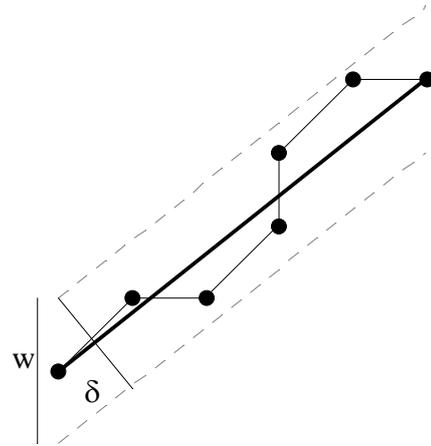


Fig. 1. Barra digital de ancho vertical w y ancho perpendicular δ .

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 ϵ 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 2ε . 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, el proceso de poligonalización sólo eliminan los puntos intermedios de segmentos de rectas digitales de pendiente múltiplo de $\pi/4$.

La figura 2 muestra aproximaciones de cuatro formas, para ejemplificar los resultados (letra a, letra m, letra c y letra f), usando tolerancias 0, 1 y 4.

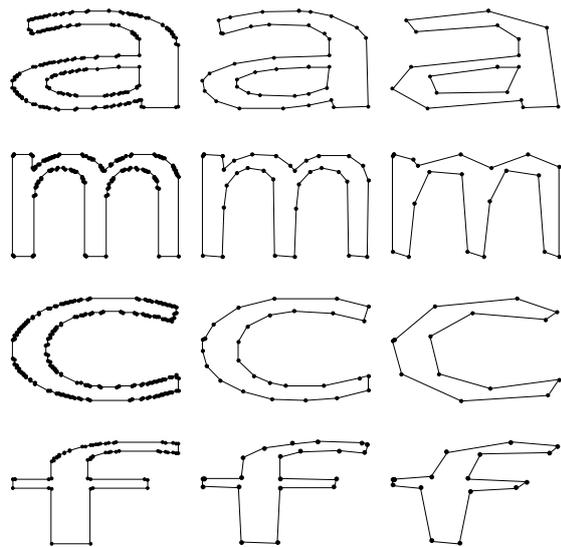


Fig. 2. Aproximación poligonal de formas de objetos con tolerancias 0, 1 y 4

2.2. Poligonalización mediante la obtención y medición de la cáscara convexa

El ancho de un conjunto de puntos se define como la menor distancia entre dos rectas paralelas que encierran a los puntos. Basándonos en estas afirmaciones, se aproximará con un segmento de recta a un conjunto de puntos con ancho menor que cierta tolerancia ε , tal como fue propuesto en [4]. La Figura 3 ilustra el proceso.

Dada la lista de puntos de borde P_0, P_1, \dots, P_n , se particiona a la lista en sublistas de puntos consecutivos $\{P_k, P_{k+1}, \dots, P_{k+m}\}$. Los puntos en cada sublista se considera que están aproximadamente alineados, esto es, el ancho del conjunto de puntos en cada sublista es menor que la tolerancia ε especificada. Esta partición se hace secuencialmente: se van analizando uno a uno los puntos del borde y se examina si puede pertenecer al lado actual, o es un punto de un lado nuevo del polígono, que comenzaría en el punto anterior. Se recibe la lista de puntos en el lado L que se está construyendo y el nuevo candidato PP_i , y mide el ancho de la cáscara convexa que contiene al conjunto de puntos formado por los puntos en L más P_i . Si el ancho de la cáscara convexa es menor que la tolerancia ε , $L = L + P_i$. Caso contrario, se inicia un nuevo lado.

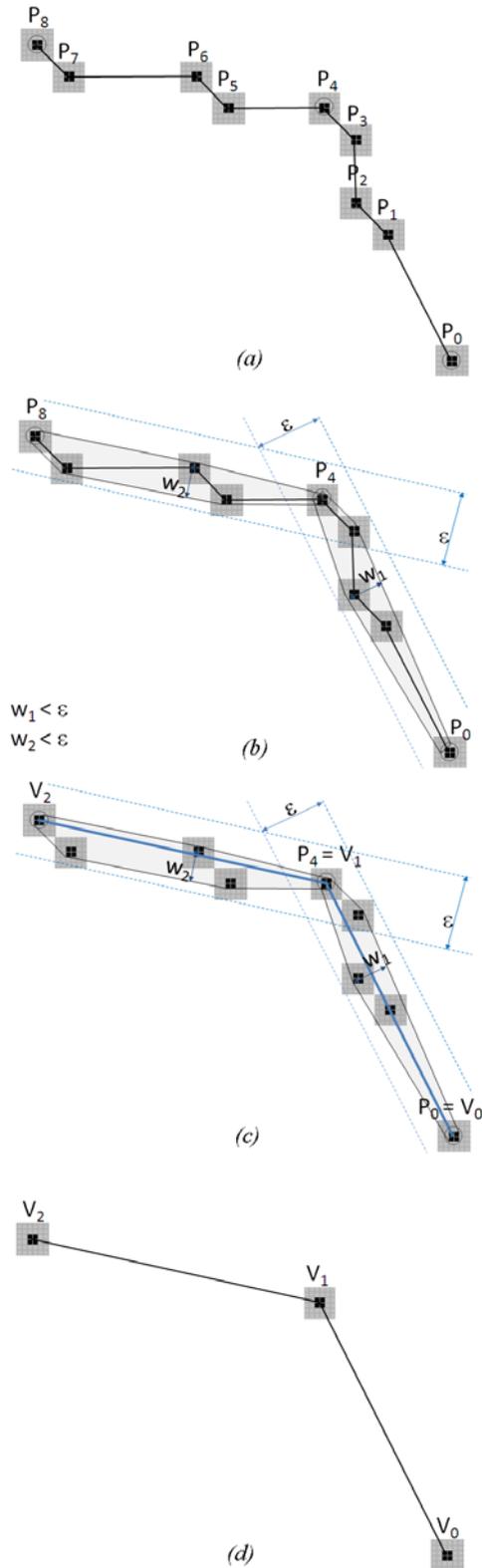


Fig. 2. Poligonalización de curvas. (a) Un conjunto de puntos. (b) Se comparación del ancho de la cascara convexa con la tolerancia ϵ . (c) Se definen los nuevos vértices. (d) Se desechan los restantes.

2.3. Breve comparación.

Se sometió a un conjunto de formas a ambos procesos de poligonalización. En general, para ambos métodos se notó que tolerancias mayores hacen perder detalles de la forma, aunque sin una notable ganancia en reducción de puntos. Sin embargo, es importante notar que el método de barras digitales implica mayor simpleza y menor costo computacional.

Tabla 1: Comparación entre los métodos de poligonalización de curvas.

Figura	Ptos. borde	Vert. de Polígonos aproximantes					
		Barras Digitales			Cásc. convexa		
		$\epsilon=0$	$\epsilon=1$	$\epsilon=4$	$\epsilon=0$	$\epsilon=1$	$\epsilon=4$
Letra a	368	163	31	13	142	28	13
Letra m	640	126	33	21	153	37	20
Letra c	368	182	21	15	176	30	13
Letra f	423	61	22	17	66	20	17

La Tabla 1 muestra el ahorro de puntos con tolerancias de 0, 1 y 4 usando ambos métodos de poligonalización.

Puede observarse que la utilización de ambos métodos presenta resultados aproximadamente iguales en referencia a la cantidad de puntos obtenidos con diferentes tolerancias.

3. El Patrón de giro: Evolución Discreta de Curvatura y Concavidad.

Es claro que el los puntos del polígono que no son vértices, la curvatura es cero y no está definida la convexidad. Por lo tanto, solo es de interés su análisis en los vértices. En esta sección se define al patrón de giro, y se describe un método para obtenerlo, previa definición de curvatura. Luego se propone un método de cálculo de la concavidad.

3.1. La Curvatura Discreta.

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

$$\kappa(s) = \left| \frac{d\mathbf{T}}{ds} \right|$$

siendo

$$\mathbf{T}(s) = \frac{(x'(s), y'(s))}{|(x'(s), y'(s))|}$$

el vector unitario tangente a la curva en el punto $\alpha(s)$. La curvatura puede pensarse como el cambio instantáneo de la inclinación del vector tangente. Es decir,

$$\kappa(s) = \lim_{h \rightarrow 0} \frac{\theta(s+h) - \theta(s)}{h}$$

con θ : dirección de \mathbf{T} .

3.2. El patrón de giro: la evolución de la curvatura

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 simples.

Sea $\beta(s)$ la curva en el plano l - κ , 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 escribirse como la gráfica de la función $\kappa_{acum}(\lambda)$, con $\lambda \in [0,1]$. Extendamos estas definiciones para el caso 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 en los puntos dentro de un segmento del polígono es 0. Sólo tiene sentido definir la curvatura en los vértices, y en este caso $\kappa(i)$ 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 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, se define

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

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 - κ , cuyos vértices son $\beta(i) = (\lambda(i), \kappa_{acum}(i))$, siendo $\lambda(i) = l(i)/L$, con $i = 0, 1, 2, \dots, n$. El punto inicial de esta curva se toma con $\lambda(0)=0, \kappa_{acum}(0)=0$.

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

En la figura 4 se muestra la representación del patrón de giro $\kappa_{acum}(\lambda)$ para tres objetos, representando la letra m, la letra c y la letra f. El patrón de la letra m (figura 4.a), indica que dicha forma tiene 4 partes convexas, que se desarrollan a partir del punto inicial en el vértice de menor ángulo, en los intervalos de longitud $[0,0.08]$, $[0.09,0.3]$, $[0.48,0.56]$ y $[0.75,0.98]$. 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 4.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 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 (figura 4.b).

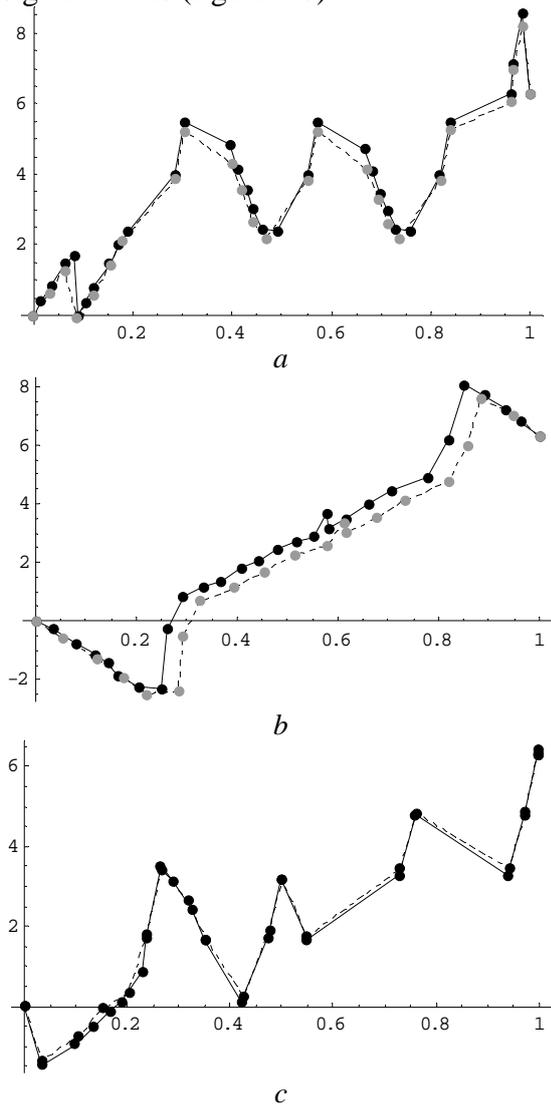


Fig. 4. Representación del patrón de giro de la poligonalización con tolerancia 1 (línea llena, puntos negros) y tolerancia 2 (línea de puntos, puntos grises). a. Letra m. b. Letra c. c. Letra f.

3.3. Cálculo de concavidad

En cada uno de los vértices del polígono es de interés el cálculo de la concavidad, a los fines de determinar cuál de los ángulos

determinados por los lados adyacentes debe ser tenido en cuenta para la estimación de la curvatura en dicho vértice.

Dados los vértices v_{i-1} , v_i , v_{i+1} , puede determinarse si en ese orden forman una “vuelta derecha” o “vuelta izquierda” [14]. Así, si el recorrido es en sentido horario, y los tres puntos determinan una vuelta izquierda, diremos que en v_i , la concavidad es positiva. En caso contrario, es negativa. En sentido antihorario, en cambio, si los tres puntos determinan una vuelta izquierda, diremos que en v_i , la concavidad es negativa. En caso contrario, es positiva.

4.2. El 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 segmentadas y no depende del métodos de poligonalización usado. 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 aplicado en este trabajo para detectar bordes 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, 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 ángulos de giro es $\tilde{\kappa}_{acum}(\lambda) = \kappa_{acum}(\lambda + \lambda_0) - \kappa_{acum}(\lambda)$, donde $\lambda_0 = l(k)/L$, y tomando κ_{acum} extendida como se explicó anteriormente. Sin embargo, obteniendo la orientación o inclinación de la forma de la curva plana no es necesaria la extensión del patrón de giro en 2π .

La detección de la orientación o ángulo de rotación se logra a partir de identificar al

menor rectángulo que encierra a la C. El ángulo que forman sus lados con el sistema de coordenadas cartesianas determina cuatro diferentes ángulos de inclinación u orientación: $\alpha + k.\pi/2$ con $k = 0, 1, 2$ o 3 y α es el ángulo de uno de los lados.

El algoritmo iniciará el recorrido del borde en solo cuatro puntos posibles de C independientemente de su orientación.

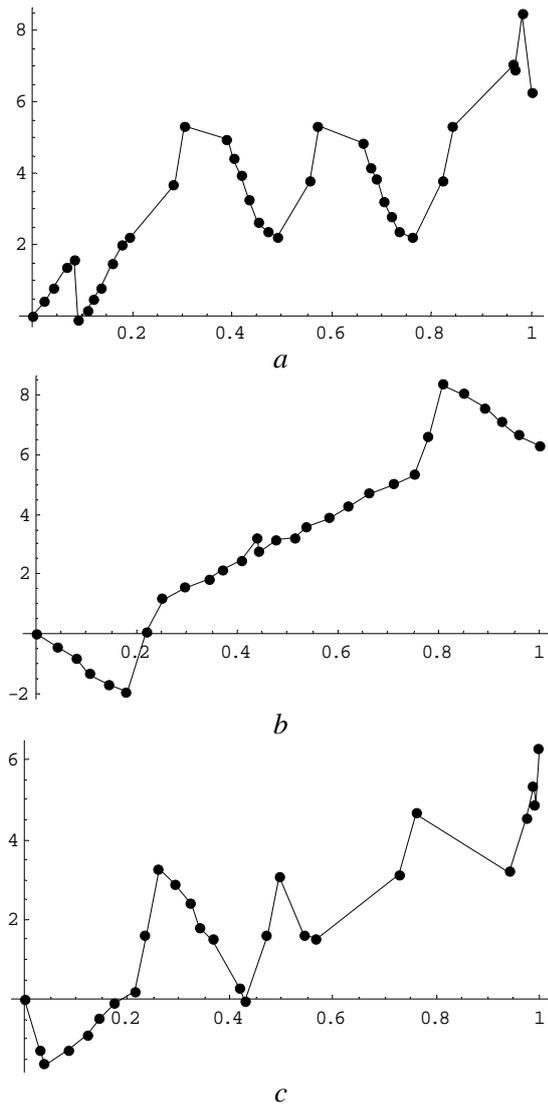


Fig. 5. El patrón de giro a partir de la poligonalización de objetos rotados con tolerancia 1. a - Letra m rotada 10° . b - Letra c rotada -45° . c - Letra f rotada -15°

A los fines de evitar el cálculo de cuatro patrones de giro para cada forma, en tiempo real durante el proceso de reconocimiento, puede realizarse el cálculo de solo uno de ellos en este proceso,

mientras que en la base de datos donde se encuentran las características de los objetos normalizados por el patrón, se almacenen los cuatro patrones de giro para los cuatro puntos. Así se asegura obtener acierto con el cálculo de solo un patrón durante el proceso de reconocimiento.

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. Tal representación fue normalizada a fin de que el primer vértice coincida con el de menor ángulo. Se observa una aceptable conformidad con el patrón obtenido en la figura 4.

4. Identificación de objetos usando el patrón de giro

En esta sección se analiza cómo puede usarse el patrón de giro (definido en la sección anterior) para la identificación de objetos en una imagen digital.

4.1. Características de la representación

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

Llamamos distancia entre dos patrones κ_1 y κ_2 al valor

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

La idea de que la representación del patrón de giro identifica una curva digital puede expresarse en los siguientes términos. Dos representaciones κ_1 y κ_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 de la forma analizada, a fin de realizar un primer filtrado entre las formas, como se describe a continuación.

Dada una forma S , y el patrón de giro $\kappa_{\text{acum}}(\lambda)$, se define la zona de concavidad,

Cv_S = conjunto de intervalos de λ en los que $\kappa_{acum}(\lambda)$ es decreciente, la zona de convexidad, Cx_S = conjuntos de intervalos de λ en los que $\kappa_{acum}(\lambda)$ es creciente. Además, se define la zona significativa recta, L_S = conjunto de intervalos maximales de λ 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.

4.2. Comparación de patrones

Sea $\kappa_{acum}(\lambda)$ el patrón de giro de una curva cuya forma S se quiere identificar. La identificación se realiza por comparación con patrones conocidos. Sean $\alpha_j(\lambda)$ patrones de formas S_j , $j=1, 2, \dots, m$ candidatas a empatarse con la forma analizada S . Así, S se identifica con S_j si

$$D(\kappa_{acum}, \alpha_j) \leq D(\kappa_{acum}, \alpha_k) \quad \forall k=1,2,\dots,m$$

Los patrones candidatos a empatar con la forma analizada se obtienen por un filtrado inicial entre todos los patrones almacenados. Este filtrado tiene en cuenta la partición del borde en zonas cóncavas y convexas, zonas rectas.

Siendo Cv_{S_j} , Cx_{S_j} y L_{S_j} las zonas cóncavas, zonas convexas y zona significativa recta de la forma S_j , la forma S_j pasa el filtrado si estos conjuntos son similares a los correspondientes a la forma analizada, S . Formalmente, se calcula

$$Cv(S, S_j) = Cv_S \cap Cv_{S_j}$$

$$Cx(S, S_j) = Cx_S \cap Cx_{S_j}$$

$$L(S, S_j) = L_S \cap L_{S_j}$$

El filtrado es positivo si se verifican las siguientes desigualdades.

$$|Cv(S, S_j)| \leq \mu |Cv_S|$$

$$|Cx(S, S_j)| \leq \mu |Cx_S|$$

$$|L(S, S_j)| \leq \mu |L_S|$$

donde $|\cdot|$ denota la longitud del conjunto, y μ es una tolerancia de similaridad.

Así, luego del filtrado, quedan las formas que tienen cierta similaridad con S , que en general es una cantidad reducida de formas. Entre éstas, la forma analizada se identifica con aquella cuya representación de patrón de giro se encuentre a menor distancia.

5. Conclusiones

La aproximación de una forma plana mediante la poligonalización de sus bordes muestra ser un recurso muy eficaz en el reconocimiento de objetos en una imagen digital. Mientras se reducen los efectos del ruido, se disminuye notoriamente la cantidad de puntos a tratar durante el proceso de análisis de su forma sin perder información relevante.

La construcción de la representación de la evolución del ángulo de giro, como extensión del concepto de la curvatura acumulada a polígonos, constituye una herramienta de gran utilidad en la identificación (o representación) de objetos. Además, la complejidad de cálculo resulta ser muy conveniente gracias a su simpleza algorítmica.

Dado que todo el proceso es simple, robusto y de muy baja complejidad, y dado que el nivel de tolerancia está en manejo del usuario, creemos que este procedimiento permitirá lograr la identificación rápida de objetos dentro de las imágenes (usando valores grandes de ϵ), o bien (con valores bajos de ϵ), ayudar a la identificación y descubrimiento de mayores detalles dentro de las imágenes digitales.

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).
- [13]. A. Rozenfeld, *Digital Topology*, *The American Mathematical Monthly*, 86(8) pp. 621 – 630, 1979.
- [14]. M. De Berg et al. *Computational geometry*, ISBN 3-540-61270-X Springer-Verlag Berlin Heidelberg New York, 1997.

Datos de Contacto:

Jorge Kamlofsky, María Bergamini.
 CAETI - Universidad Abierta Interamericana. Av.
 Montes de Oca 725 – Buenos Aires – Argentina.
 {jorge.kamlofsky, maria.bergamini}@uai.edu.ar