

Reconhecimento de Formas Planas Invariante sob Transformação Afim

Resumo Um objeto visto de diferentes pontos de vista resulta em imagens diferentemente deformadas. O reconhecimento de formas invariante sob transformação afim deve classificar corretamente um objeto, independentemente do ponto de vista. Neste trabalho, propomos novas características locais e globais invariante sob transformação afim. Estas características podem ser usadas para reconhecimento supervisionado ou não-supervisionado de formas, e para a recuperação e indexação de imagens baseado no contorno da forma. Uma das características propostas está relacionada com a deficiência convexa e as outras são extraídas a partir da matriz de áreas. A matriz de áreas foi usada por Shen [20] para o casamento por similaridade na recuperação de imagens. Este casamento é realizado entre duas matrizes para determinar a similaridade das formas. Porém, diferentemente do trabalho de Shen, reduzimos a informação contida na matriz de áreas em três vetores coluna. Além disso, parametrizamos o contorno das formas usando o parâmetro comprimento de arco afim. Isto torna as nossas características robustas a parametrização da forma, enquanto que o trabalho de Shen não possui esta propriedade. Os resultados experimentais indicam que as nossas características podem classificar corretamente até as formas altamente deformadas e formas ruidosas usando um pequeno conjunto de treinamento.

Keywords Shape · Features invariant · Affine transformation · Affine length · Affine parameterization

1 Introduction

Um problema importante no reconhecimento de objetos é o fato de que um objeto pode ser visto desde diferentes pontos de vista, resultando em imagens diferentes. Conseqüentemente, a invariância a ponto de vista é uma propriedade desejável em muitos sistemas de reconhecimento de formas. Para objetos quase planos, estas deformações podem

ser modeladas aproximadamente por transformações afins, se os pontos de vista estiverem suficientemente distantes do objeto. Na literatura, encontramos muitos trabalhos sobre o reconhecimento de formas invariante sob transformação afim. Alguns exemplos são os descritores de Fourier invariantes afins [1,4], os momentos invariantes afins [6,8,22] e o espaço-escala da curvatura afim [12,13]. Nestes trabalhos, a idéia fundamental é realizar uma parametrização invariante afim do contorno das formas, usualmente isto é feito através do uso do parâmetro comprimento de arco afim [1,6,19,22].

Usualmente, uma boa taxa de classificação pode ser obtida usando simultaneamente as características locais e globais. As características locais são sensíveis a ruídos e, na prática, a quantidade de informações locais disponíveis são usualmente insuficientes para uma classificação robusta. Alguns exemplos de características locais são os invariantes geométricos [18], as esquinas (corners) [17] e a curvatura [11]. Por outro lado, as características globais são relativamente insensíveis a ruídos e duas formas similares normalmente possuem as características globais semelhantes. Dois exemplos simples de características globais são a área e a deficiência convexa.

A deficiência convexa é uma característica global que contém informação sobre as concavidades da forma. Esta característica é invariante sob transformação de translação, rotação e facilmente pode se tornar invariante à escala uniforme. Não menos simples é mostrar que ela pode também se tornar invariante afim. Na literatura, não encontramos nenhum resultado sobre esta propriedade da deficiência convexa. Neste trabalho, mostramos uma forma de tornar esta característica invariante sob transformação afim. As outras características propostas são extraídas a partir da matriz de áreas. A matriz de áreas, obtida a partir do contorno de uma forma, foi usada por Shen [20] para o casamento de formas por similaridade em recuperação de imagens invariante afim. Este casamento é realizado entre duas matrizes. A matriz de áreas contém muita informação sobre a forma. Para ser útil, diferentemente do trabalho de Shen, reduzimos a informação contida na matriz de áreas em três vetores coluna. A partir destes vetores, definimos as outras características. Shen descreve duas propriedades interessantes da matriz de áreas. A primeira é que ela pode se tornar in-

variante sob transformação afim. A segunda é que as suas colunas contém informações multi-escala: as colunas centrais contém informações globais ou em escala grossa e as colunas periféricas contém informações locais ou em escala fina. Diferentemente do trabalho de Shen, parametrizamos as formas usando o parâmetro comprimento de arco afim. Este procedimento torna a matriz de áreas robusta a parametrização da forma. A complexidade computacional do trabalho de Shen depende da dimensão da matriz de áreas. Nós, ao contrário, extraímos e processamos somente as informações relevantes, diminuindo a complexidade computacional. Na fase de reconhecimento, usamos o algoritmo dos k -vizinhos mais próximos.

O resto do trabalho está organizado como segue. A Seção 2 trata das derivadas discretas, da área, da transformação afim e da parametrização afim e sua implementação. Na Seção 3, descrevemos a matriz de áreas, a sua normalização e a extração de informações relevantes. A Seção 4 trata das características propostas e sua análise, do algoritmo de treinamento e do algoritmo de classificação. Os resultados experimentais estão expostos na Seção 5. Finalmente, na Seção 6 apresentamos as conclusões deste trabalho.

2 Conceitos Fundamentais

Seja $\mathbf{u} : [0, T] \rightarrow \mathbb{R}^2$ uma curva paramétrica fechada, de período T , ou seja:

$$\mathbf{u}(t) = (x(t), y(t)) \quad \text{para } t \in [0, T]. \quad (1)$$

Observe que existem infinitas curvas paramétricas geometricamente equivalentes a \mathbf{u} . Como consequência, uma mesma curva pode ter diferentes parametrizações. Mais ainda, estas parametrizações mudam sob transformações afins.

Em geral, as curvas de uma aplicação podem ter sido parametrizadas com um número diferente de pontos T . Usualmente, todas as curvas são normalizadas para ter o mesmo número de pontos. Este número de pontos não pode ser muito pequeno (perda de informação na representação do contorno da forma) ou muito grande (aumento da complexidade computacional). Na prática, este número de pontos é estimado experimentalmente para uma aplicação em particular.

2.1 Derivadas Discretas

Nesta subseção, mostramos brevemente três formas de calcular numericamente uma aproximação da primeira e da segunda derivada de uma função. Estas aproximações são baseadas nas equações de diferenças finitas da função os quais são obtidas a partir da expansão em série de Taylor da função a ser aproximada. Uma referência mais detalhada e completa encontra-se em [7].

A primeira e segunda derivadas de $\mathbf{u}(t)$ são denotadas, respectivamente, como:

$$\dot{\mathbf{u}}(t) = (\dot{x}(t), \dot{y}(t)), \quad \ddot{\mathbf{u}}(t) = (\ddot{x}(t), \ddot{y}(t)).$$

Lembremos que temos somente a seqüência de pontos $\mathbf{u}(0), \mathbf{u}(1), \dots, \mathbf{u}(T)$ a partir dos quais devemos calcular as derivadas. Nas seguintes aproximações, o valor de h é constante e na prática considera-se $h = 1$.

Forward Differences

$$\dot{\mathbf{u}}(t) = \frac{\mathbf{u}(t+h) - \mathbf{u}(t)}{h}$$

$$\ddot{\mathbf{u}}(t) = \frac{\mathbf{u}(t+2h) - 2\mathbf{u}(t+h) + \mathbf{u}(t)}{h^2}$$

Backward Differences

$$\dot{\mathbf{u}}(t) = \frac{\mathbf{u}(t) - \mathbf{u}(t-h)}{h}$$

$$\ddot{\mathbf{u}}(t) = \frac{\mathbf{u}(t) - 2\mathbf{u}(t-h) + \mathbf{u}(t-2h)}{h^2}$$

Central Differences

$$\dot{\mathbf{u}}(t) = \frac{\mathbf{u}(t+h) - \mathbf{u}(t-h)}{2h}$$

$$\ddot{\mathbf{u}}(t) = \frac{\mathbf{u}(t+h) - 2\mathbf{u}(t) + \mathbf{u}(t-h)}{h^2}$$

2.2 Transformação Afim

Uma transformação afim $\mathbf{F} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ atuando sobre a curva \mathbf{u} é uma aplicação tal que [21]:

$$\mathbf{F}(\mathbf{u}) = \mathbf{A}\mathbf{u} + \mathbf{T} \quad (2)$$

onde \mathbf{A} é uma matriz 2×2 inversível e $\mathbf{T} \in \mathbb{R}^2$ é o vetor de translação. Daqui em diante chamaremos a tal matriz \mathbf{A} de *matriz afim*. Se $|\mathbf{A}| > 0$, então a orientação das curvas \mathbf{u} e $\mathbf{F}(\mathbf{u})$ é a mesma, caso contrário as curvas tem diferentes orientações.

As transformações de rotação, translação e mudança de escala uniforme não alteram a descrição geométrica da forma. Por outro lado, as transformações de escala não-uniforme e inclinação deformam a forma. A transformação afim contém todos estes tipos de transformações. As seguintes matrizes representam, respectivamente, as transformações de escala, rotação e shearing:

$$A_s = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} A_\theta = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} A_\kappa = \begin{bmatrix} 1 & \kappa \\ 0 & 1 \end{bmatrix} \quad (3)$$

Se $s_x = s_y$ então a matriz A_s representa a transformação de escala uniforme, caso contrario representa a transformação de escala não-uniforme. Na matriz A_κ , o parâmetro κ , chamado de taxa de deformação (shear ratio), controla a medida da deformação.

2.3 Área e Transformação Afim

Seja Δ o triângulo formado pelos pontos $v_1 = (x_1, y_1)$, $v_2 = (x_2, y_2)$ e $v_3 = (x_3, y_3)$. Então, a área do triângulo Δ é dada pela seguinte fórmula [15]:

$$\text{Area}(\Delta) = \frac{1}{2} \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} \quad (4)$$

Se v_1 , v_2 e v_3 são orientados no sentido anti-horário o valor da área é positivo, caso contrário é negativo. Se \mathbf{F} é uma transformação afim e \mathbf{A} a sua matriz afim associada, então:

$$\text{Area}(\mathbf{F}(\Delta)) = |\mathbf{A}| \text{Area}(\Delta) \quad (5)$$

onde $|\mathbf{A}|$ é o determinante da matriz \mathbf{A} . Considerando a Equação 5 e o fato de que qualquer polígono simples 2-D \mathbf{u} pode ser decomposto como um conjunto de triângulos, podemos afirmar o seguinte:

$$\text{Area}(\mathbf{F}(\mathbf{u})) = |\mathbf{A}| \text{Area}(\mathbf{u}) \quad (6)$$

Neste trabalho, o fecho convexo de uma curva paramétrica fechada \mathbf{u} é definido como a seqüência de pontos que descrevem o menor polígono convexo contendo \mathbf{u} . Denotamos o fecho convexo da curva \mathbf{u} como $\text{FC}(\mathbf{u})$. Para o cálculo do fecho convexo de \mathbf{u} , usamos o Algoritmo de Graham [16]. Neste algoritmo, a primitiva é o sinal da área do triângulo que verifica se um ponto está à esquerda de um vetor. Este sinal não muda sob transformação afim de \mathbf{u} . Assim, podemos afirmar que um ponto pertence ao $\text{FC}(\mathbf{u})$ se, e somente se, a sua transformada pertencer ao $\text{FC}(\mathbf{F}(\mathbf{u}))$. Conseqüentemente, podemos também afirmar o seguinte:

$$\text{Area}(\text{FC}(\mathbf{F}(\mathbf{u}))) = |\mathbf{A}| \text{Area}(\text{FC}(\mathbf{u})) \quad (7)$$

Baseado nas Equações 6 e 7, observamos que a deficiência convexa é invariante sob transformação afim a menos do fator de proporcionalidade $|\mathbf{A}|$. Este fator pode ser simplificado dividindo a deficiência convexa, por exemplo, pela área do contorno ou pela área do fecho convexo do contorno. Neste trabalho, propomos e definimos a nossa primeira característica invariante afim da seguinte forma:

$$F_1 = \frac{\text{Area}(\text{FC}(\mathbf{u})) - \text{Area}(\mathbf{u})}{\text{Area}(\text{FC}(\mathbf{u}))} \quad (8)$$

2.4 Parametrização Afim

Na literatura, encontramos dois parâmetros invariantes usados para parametrizar curvas planas e úteis para resolver satisfatoriamente problemas de reconhecimento de formas. Estes parâmetros são o comprimento de arco e comprimento de arco afim.

O parâmetro comprimento de arco é invariante sob transformação de similaridade [14]. Porém, ele não é invariante sob transformação afim. O comprimento de arco normalizado da curva \mathbf{u} é definido como:

$$s(p) = \frac{\int_0^p (\dot{x}^2(t) + \dot{y}^2(t))^{\frac{1}{2}} dt}{\int_0^T (\dot{x}^2(t) + \dot{y}^2(t))^{\frac{1}{2}} dt} \quad (9)$$

Para resolver satisfatoriamente problemas de reconhecimento de formas invariante sob transformação afim, é necessário re-parametrizar as formas através de um parâmetro invariante afim. Neste trabalho, usamos o parâmetro comprimento de arco afim [6, 12, 19, 22]. O comprimento de arco afim normalizado da curva \mathbf{u} é definido da seguinte forma:

$$s_a(p) = \frac{\int_0^p |\dot{x}(t)\ddot{y}(t) - \ddot{x}(t)\dot{y}(t)|^{\frac{1}{3}} dt}{\int_0^T |\dot{x}(t)\ddot{y}(t) - \ddot{x}(t)\dot{y}(t)|^{\frac{1}{3}} dt} \quad (10)$$

Na implementação da parametrização afim do contorno de uma forma, é desejável que o parâmetro s_a seja uma função estritamente crescente (isto é, não existe p_i e p_j tal que $s_a(p_i) = s_a(p_j)$). Para garantir isto, removemos todos os pontos da curva \mathbf{u} com a segunda derivada nula. Isto equivale a remover os pontos pertencentes a linhas retas.

Neste trabalho, normalizamos o número de pontos de todas as formas, pois as características propostas dependem da escolha deste número. Também re-parametrizamos todas as formas usando o parâmetro comprimento de arco afim normalizado. Desta forma, as nossas características invariantes afins serão robustas a parametrização da forma.

2.5 Implementação da Parametrização Invariante

Nesta subseção, descrevemos os passos principais para a implementação da parametrização invariante de formas. Especificamente, estes passos se resumem em:

1. Dado o contorno $\mathbf{u}(t)$, calcular o parâmetro invariante $\eta(p)$, $p = 0, \dots, T$;
2. Dado o número de pontos para a re-parametrização, dividir o parâmetro $\eta(p)$ em δs partes iguais;
3. Interpolar p para obter \bar{p} tal que $\eta(\bar{p})$ seja uma função dividida em δs partes iguais;
4. Interpolar o contorno $\mathbf{u}(t)$ nos pontos $\eta(\bar{p})$ para obter a re-parametrização do contorno.

Na interpolação, pode-se usar qualquer método conhecido: linear, cúbico, spline, etc. Neste trabalho escolhemos o método linear. Observe que o contorno re-parametrizado adquire uma propriedade baseada na propriedade do parâmetro $\eta(\bar{p})$. No caso do parâmetro comprimento de arco normalizado, os pontos consecutivos sobre o contorno são equidistantes. No caso do parâmetro comprimento de arco afim normalizado, a área do triângulo formado por três pontos consecutivos sobre o contorno é constante.

Considere o contorno ilustrado na Figura 1(a) como o contorno a ser re-parametrizado pelos parâmetros comprimento de arco normalizado e comprimento de arco afim normalizado. Este contorno consiste de 20 pontos ($t = 1, 2, \dots, 20$). A Figura 2 ilustra estes parâmetros divididos em partes iguais.

A Figura 1(b) (Figura 1(c)) ilustra a re-parametrização do contorno pelo parâmetro comprimento de arco normalizado (comprimento de arco afim normalizado). No caso da re-parametrização pelo comprimento de arco normalizado, os pontos sobre o contorno são bastante equidistantes. Porém, o contorno perdeu informação sobre detalhes

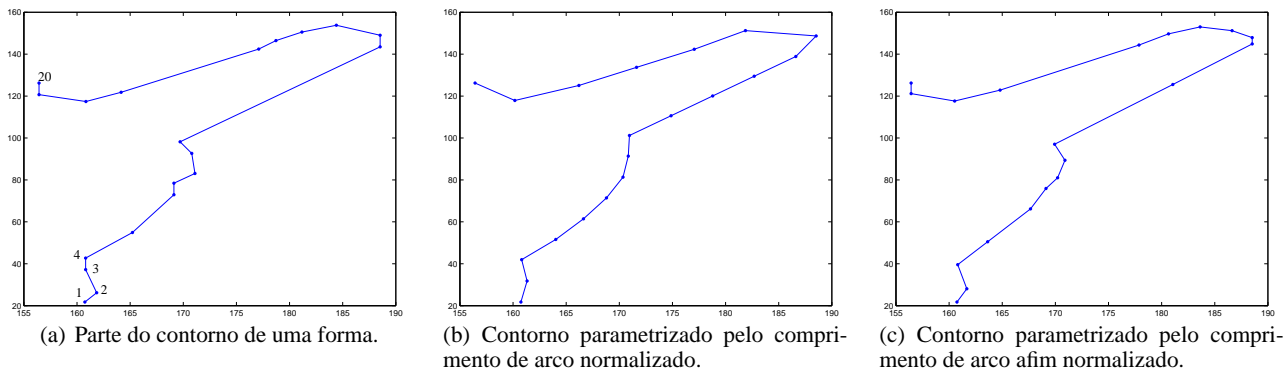


Figura 1 Contorno original e duas re-parametrizações

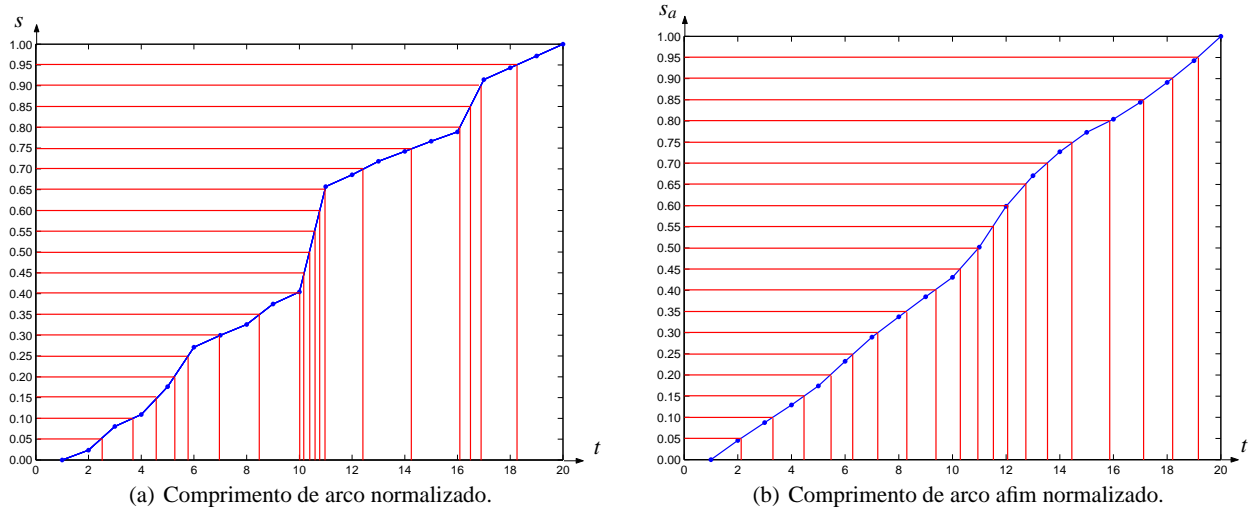


Figura 2 Parâmetros invariantes divididos em partes iguais.

pequenos. No caso da re-parametrização pelo comprimento de arco afim normalizado, a propriedade do triângulo com área constante não se cumpre bem para todos os pontos. Isto é devido a que o cálculo do parâmetro invariante afim envolve o uso de derivadas de segundo ordem e como consequência o ruído é maior. Porém, o contorno não perdeu informação relevante. Finalmente, para mostrar o resultado da implementação da parametrização sobre um contorno maior, a Figura 3 ilustra as re-parametrizações de um contorno de gesto da mão.

3 Extração de Informações Locais e Globais de uma Matriz de Áreas

Antes de definir a matriz de áreas do contorno de uma forma, mostramos a seguinte proposição que servira para entender a natureza e as propriedades da matriz de áreas.

Proposition 1 Seja $\mathbf{u}(t) = (x(t), y(t))$ uma curva paramétrica no plano com parametrização arbitrária t . Então,

$\Delta(t) = \dot{x}(t)\ddot{y}(t) - \ddot{x}(t)\dot{y}(t)$ é proporcional à área do triângulo formado por três pontos consecutivos do contorno \mathbf{u} .

Considerando as diferenças centrais como aproximação para a primeira e segunda derivada de \mathbf{u} , obtemos o seguinte:

$$\dot{x}(t) = \frac{x(t+h) - x(t-h)}{2h} \quad (11)$$

$$\ddot{x}(t) = \frac{x(t+h) - 2x(t) + x(t-h)}{h^2} \quad (12)$$

Similarmente, podemos obter as aproximações para $\dot{y}(t)$ e $\ddot{y}(t)$. Desta forma, o valor de $\Delta(t)$ pode ser expressado como:

$$\begin{aligned} \Delta(t) &= \frac{1}{h^3} \begin{vmatrix} x(t-h) & y(t-h) & 1 \\ x(t) & y(t) & 1 \\ x(t+h) & y(t+h) & 1 \end{vmatrix} \\ &= \frac{2}{h^3} \text{Area}(\mathbf{u}(t-h), \mathbf{u}(t), \mathbf{u}(t+h)) \end{aligned} \quad (13)$$

Se $h = 1$ então $\Delta(t)$ é proporcional a área do triângulo formado pelos pontos consecutivos $\mathbf{u}(t-1)$, $\mathbf{u}(t)$ e $\mathbf{u}(t+1)$.

O sinal da curvatura de $\mathbf{u}(t)$ é igual ao sinal de $\Delta(t)$. Para $h = 1$, a Equação 13 mostra que este sinal também é igual

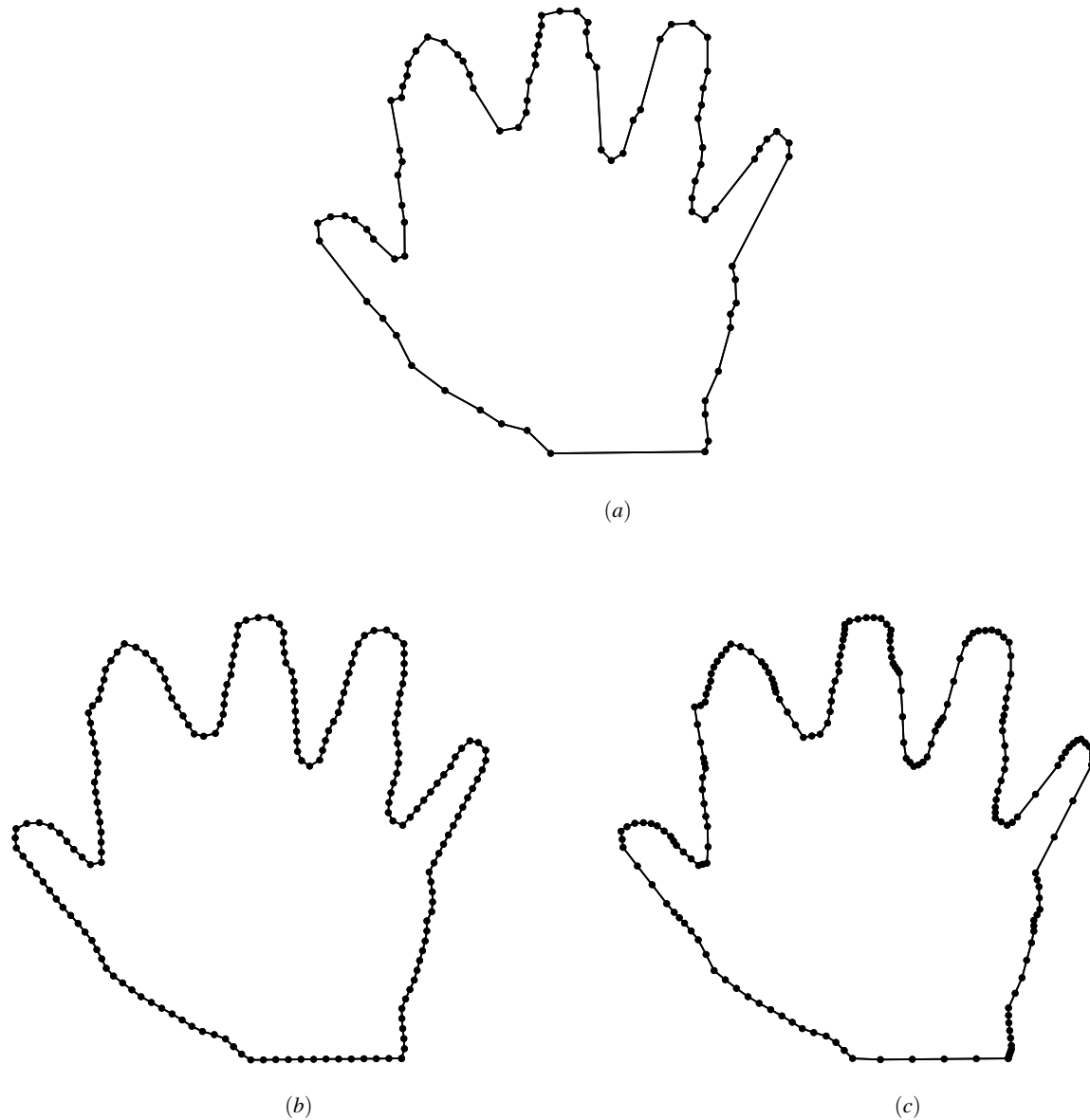


Figura 3 Algumas re-amostragens de uma curva: (a) contorno original com parametrização arbitrária (89 pontos), (b) parametrização pelo comprimento de arco normalizado (200 pontos), e (c) parametrização pelo comprimento de arco afim normalizado (200 pontos).

ao sinal da $\text{Area}(\mathbf{u}(t-1), \mathbf{u}(t), \mathbf{u}(t+1))$. O valor deste sinal nos indica as partes côncavas e convexas do contorno \mathbf{u} .

Portanto, considerando h fixo e $t = 0, 1, \dots, T-1$, a seguinte curva nos fornece informações sobre as partes côncavas e convexas do contorno \mathbf{u} :

$$\Psi_h(t) = \text{Area}(\mathbf{u}(t-h), \mathbf{u}(t), \mathbf{u}(t+h)) \quad (14)$$

A proposição anterior mostra a relação entre Ψ_h e a curvatura. Observe que a função Ψ_h é invariante sob transformação afim do contorno \mathbf{u} a menos de um fator de proporcionalidade.

A curvatura é um dos mais importantes atributos que podem ser extraídos de um contorno [3,5]. Motivados pela

relação que existe entre Ψ_h e a curvatura, nos exploramos a função Ψ_h para $h \geq 1$. Considerando $h = 1, 2, 3, \dots$ teremos um conjunto de curvas $\Psi_1, \Psi_2, \Psi_3, \dots$ que descrevem informações sobre as partes côncavas e convexas do contorno \mathbf{u} numa “escala” h . A Figura 4 ilustra estas curvas obtidos a partir da forma ilustrado na Figura 3(a).

Esta forma de descrever uma curva em diferentes “escalas” foi também usada por N. Arica [2] para descrever a curvatura. Considerando a curvatura em todas as “escalas” possíveis, ele definiu o descritor BAS (Beam Angle Statistics), invariante sob transformação de similaridade, com excelente desempenho sobre o Core Experiment CE-Shape-1 [9].

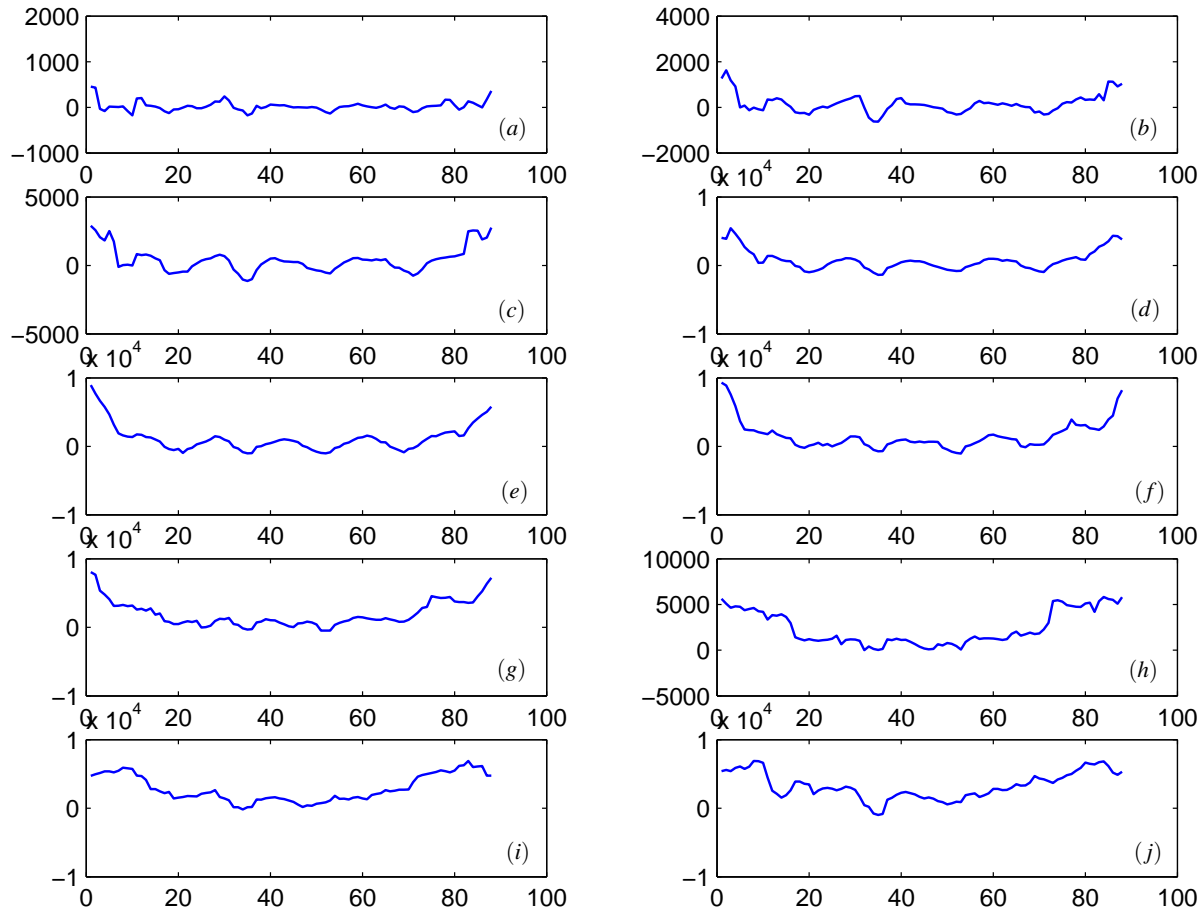


Figura 4 As curvas Ψ_h .

3.1 Matriz de Áreas

Shen [20] define a matriz de áreas de uma curva \mathbf{u} como sendo a matriz que tem como vetores coluna as curvas Ψ_h . Seja $\mathbf{u}(t)$ ($t = 0, \dots, T$), uma curva fechada de período T e seja $m = (T - 1)/2$. Seja \mathbf{S}_{ti} a área do triângulo formado pelos pontos $\mathbf{u}(t-i)$, $\mathbf{u}(t)$ e $\mathbf{u}(t+i)$. Então, a matriz de áreas de $\mathbf{u}(t)$ é definida como:

$$\mathbf{S} = \begin{bmatrix} \mathbf{S}_{01} & \mathbf{S}_{02} & \mathbf{S}_{03} & \cdots & \mathbf{S}_{0m} \\ \mathbf{S}_{11} & \mathbf{S}_{12} & \mathbf{S}_{13} & \cdots & \mathbf{S}_{1m} \\ \mathbf{S}_{21} & \mathbf{S}_{22} & \mathbf{S}_{23} & \cdots & \mathbf{S}_{2m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{S}_{(T-1)1} & \mathbf{S}_{(T-1)2} & \mathbf{S}_{(T-1)3} & \cdots & \mathbf{S}_{(T-1)m} \end{bmatrix} \quad (15)$$

Uma simples análise sobre a matriz de áreas nos mostra que ela contém informação local à esquerda, global no centro e local à direita:

$$\mathbf{S} = \begin{bmatrix} \mathbf{S}_{t1} \cdots \mathbf{S}_{tk} & \mathbf{S}_{t(k+1)} \cdots \mathbf{S}_{tl} & \mathbf{S}_{t(l+1)} \cdots \mathbf{S}_{tm} \end{bmatrix}^{t=0, \dots, T-1}$$

informação local informação global informação local

O i -ésimo vetor coluna de \mathbf{S} contém informações locais (escala fina) quando o índice i for pequeno ou próximo a m e contém informações globais (escala grossa) quando i estiver

situado na região central de \mathbf{S} . De fato, cada vetor coluna da matriz de áreas analisa a forma numa determinada escala, e a matriz de áreas completa contém as informações multi-escala da forma. Observemos também que a magnitude das informações locais são muito menores que a magnitude das informações globais.

3.2 Matriz de Áreas Normalizada

A matriz de áreas contém informações multi-escala. Para ser útil, devemos diminuir a dimensão desta matriz. No trabalho de Shen [20], a similaridade das formas é calculada através do casamento de duas matrizes. Diferentemente do trabalho de Shen, reduzimos a informação contida na matriz de áreas em três vetores coluna. Antes que a matriz de áreas seja reduzida, ela deve ser normalizada: tornar a matriz invariante sob transformação afim e ter todas as colunas com magnitudes similares.

Normalizamos a matriz de áreas \mathbf{S} em dois passos. Primeiro, tornamos a matriz \mathbf{S} invariante sob transformação afim dividindo cada elemento de \mathbf{S} pela soma de todos os elementos de \mathbf{S} . Segundo, fazemos todas as colunas terem magnitudes similares: Dividimos cada coluna i de \mathbf{S} pelo

Tabela 1 Configuração das três gaussianas (m é o número de colunas de \mathbf{S}).

Gaussiana	σ	μ	Δm
g_e	5	$0.20m$	$0.20m$
g_c	5	$0.50m$	$0.15m$
g_d	5	$0.85m$	$0.15m$

peso $w(i)$. O vetor de pesos w é calculado a partir de uma matriz de áreas invariante afim \mathbf{R} de uma forma representativa usando a seguinte equação:

$$w(i) = \max_{t=0, \dots, T-1} \{|\mathbf{R}_{t,i}|\} \quad i = 1, \dots, m \quad (16)$$

O método proposto para fazer todas as colunas terem magnitudes similares não é a melhor, porém serve para os nossos propósitos.

3.3 Extração de Informações Locais e Globais

Uma matriz de áreas contém informações em quantidade excessiva. Para ser útil, ela precisa ser reduzida e organizada de alguma forma. Extraímos as informações relevantes a partir de uma matriz de áreas normalizada \mathbf{S} e as armazenamos em três vetores ψ_e , ψ_c e ψ_d que resumem, respectivamente, as informações das colunas situadas à esquerda, no centro e à direita da matriz \mathbf{S} :

$$\mathbf{S} = \begin{bmatrix} \mathbf{S}_{i1} \cdots \mathbf{S}_{ik} & \mathbf{S}_{i(k+1)} \cdots \mathbf{S}_{il} & \mathbf{S}_{i(l+1)} \cdots \mathbf{S}_{im} \\ \text{informação local} & \text{informação global} & \text{informação local} \end{bmatrix}_{t=0, \dots, T-1}$$

$$\begin{matrix} \downarrow & \downarrow & \downarrow \\ \psi_e & \psi_c & \psi_d \end{matrix}$$

Neste trabalho, propomos o seguinte método para calcular cada um dos três vetores. Por exemplo, o vetor ψ_e é calculado da seguinte forma:

$$\psi_e(t) = \frac{\sum_i g_e(i, \mu, \sigma) \mathbf{S}_{t,i}}{\sum_i g_e(i, \mu, \sigma)} \quad t = 0, \dots, T-1 \quad (17)$$

onde g_e é uma curva gaussiana. O centro μ desta gaussiana está situado no lado esquerdo de \mathbf{S} . Ele não pode estar muito próximo à primeira coluna, para reduzir os ruídos presentes em escala finas. O desvio-padrão σ é escolhido para extrair todas as informações presentes no lado esquerdo de \mathbf{S} , sem que haja mistura com as informações da região central de \mathbf{S} . Na implementação da curva gaussiana, consideramos que ela está definida somente no intervalo $[\mu - \Delta m, \mu + \Delta m]$ como mostrado na Figura 5(a). Os vetores ψ_c e ψ_d são calculados de forma similar.

Os parâmetros que definem as três gaussianas são definidos experimentalmente para uma aplicação em particular. Embora, esses parâmetros poderiam ser úteis também para outras aplicações. Neste trabalho, as três gaussianas foram definidas de acordo a Tabela 1.

Se consideramos curvas fechadas com $T = 140$ pontos, então a Figura 5(b) ilustra a configuração das três gaussianas (neste caso $m = 69$).

4 Características e Algoritmo de Classificação

A seguir, definimos as nossas características invariante sob transformações afins. Elas são definidas a partir dos vetores ψ_e , ψ_c e ψ_d . O algoritmo de treinamento e classificação também são apresentados.

4.1 Características Propostas

Os vetores ψ_e , ψ_c e ψ_d foram definidos na Subseção 3.3. Usamos cinco características para classificar as formas:

- A característica F_1 foi definida na Equação 8:

$$F_1 = \frac{\text{Area}(\text{FC}(\mathbf{u})) - \text{Area}(\mathbf{u})}{\text{Area}(\text{FC}(\mathbf{u}))}$$

- A característica F_2 é a soma dos valores negativos de ψ_e :

$$F_2 = \sum_{\psi_e(t) < 0} \psi_e(t)$$

- A característica F_3 é a soma dos valores positivos de ψ_e :

$$F_3 = \sum_{\psi_e(t) > 0} \psi_e(t)$$

- A característica F_4 é a soma de ψ_c :

$$F_4 = \sum \psi_c(t)$$

- A característica F_5 é a soma de ψ_d :

$$F_5 = \sum \psi_d(t)$$

No início da escolha das características, começamos testando as somas das partes negativas e positivas dos três vetores sob ruído e deformação. Os resultados foram satisfatórios somente para o vetor ψ_e .

Observe que acabamos reduzindo a enorme quantidade de informação contida numa matriz de áreas em cinco características (vetor de características). Diferentemente do trabalho de Shen [20], que mede a similaridade de duas formas através do casamento de duas matrizes, vamos medir a similaridade de duas formas através da distância entre seus vetores de características. Isto torna muito mais rápido o processo de classificação. Além disso, parametrizamos o contorno das formas usando o parâmetro comprimento de arco afim. Isto torna as nossas características robustas a parametrização da forma, enquanto que o trabalho de Shen não possui esta propriedade.

Todas estas características são invariantes sob transformação afim. Elas também são invariantes à escolha do ponto inicial. Usamos o algoritmo dos k -vizinhos mais próximos para classificar as formas. Para poder usar este algoritmo, normalizamos as características para terem a média zero e o desvio-padrão um.

4.2 Análise das Características

Nesta subseção, apresentamos uma análise das características propostas. Esta análise é feita a partir dos vetores

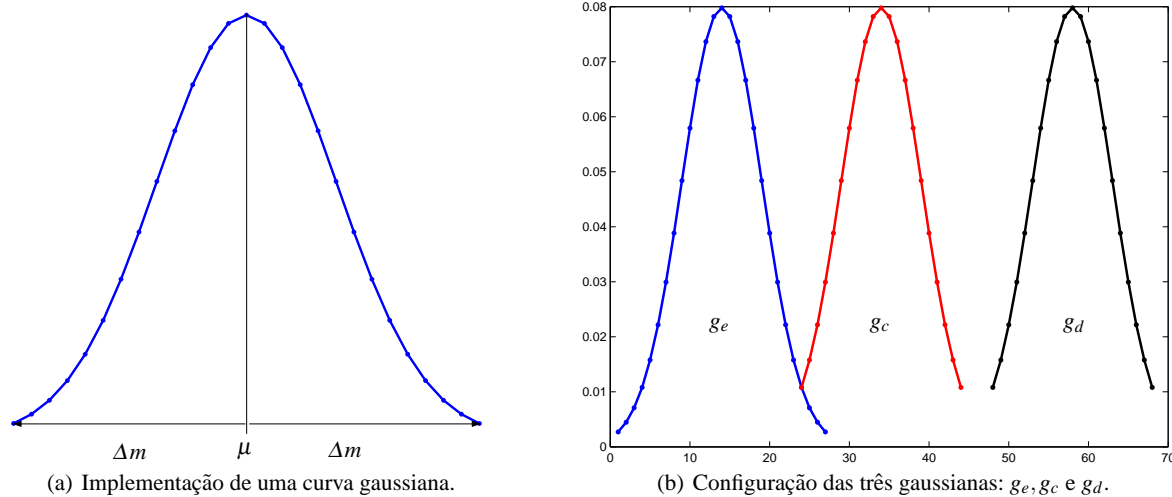


Figura 5 Escolhemos a curva gaussiana com o objetivo de reduzir ruído e não misturar informações locais com informações globais.

ψ_e , ψ_c e ψ_d . Especificamente, estudamos a mudança dos vetores para diferentes parametrizações de uma mesma forma, a robustez dos vetores para formas similares e a invariância dos vetores sob transformação de inclinação.

Os três vetores são baseados na soma de área de triângulos formados por três pontos pertencentes ao contorno da forma. Assim, eles são altamente dependentes a parametrização da forma. A Figura 6 ilustra este fato para uma mesma forma parametrizada pelo parâmetro comprimento de arco (forma (a)) e comprimento de arco afim (forma (c)). Nesta figura, (b) e (d) são as curvas que representam os vetores ψ_e , ψ_c e ψ_d para os contornos (a) e (c), respectivamente. Observe que a dependência da parametrização da forma é maior para ψ_c e ψ_d .

Para duas formas similares e parametrizadas através do parâmetro comprimento de arco afim, as curvas que representam os vetores ψ_e , ψ_c e ψ_d também resultam similares. Isto é ilustrado na Figura 7. Nesta figura, (a) e (c) representam duas formas similares parametrizadas pelo parâmetro comprimento de arco afim, (b) e (d) são as curvas que representam os três vetores para os contornos (a) e (c), respectivamente. Observe que o vetor ψ_e é o mais robusto.

A invariância das curvas que representam os vetores ψ_e , ψ_c e ψ_d sob transformação de inclinação é ilustrado na Figura 8. Os contornos deformados ((b), (c) e (d)) são obtidos a partir da aplicação de uma transformação de inclinação ($\kappa = 1, 2$ e 3) sobre o contorno original (a). Antes de extrair os três vetores, parametrizamos o contorno original e os contornos deformados através do parâmetro comprimento de arco afim. Observe que os três vetores são os mesmos para o contorno original e deformados. A prova disto está no fato de que o parâmetro comprimento de arco afim e os vetores ψ_e , ψ_c e ψ_d são invariantes sob transformações afins.

4.3 Algoritmo de Treinamento

O algoritmo de treinamento consiste em calcular as cinco características e deve ser aplicado a todos os exemplos de treinamento. Cada exemplo de treinamento deve ser apresentado a este algoritmo junto com a sua correta classificação:

- Normalize o número de pontos da forma.
- Re-parametrize a forma usando o parâmetro comprimento de arco afim (Equação 10).
- Calcule e normalize a matriz de áreas.
- Determine os vetores ψ_e , ψ_c e ψ_d .
- Calcule as cinco características da forma.

4.4 Algoritmo de Classificação

O algoritmo de classificação recebe como entrada uma forma de consulta com classificação desconhecida e o número k a ser usado no algoritmo dos k -vizinhos mais próximos. A saída deste algoritmo é a classificação da forma.

- Dada uma forma de consulta, calcule as cinco características como descrito no algoritmo de treinamento. Denotemos o vetor de características obtido como x .
- Procure, no conjunto de treinamento, k formas que estão situadas mais próximas ao vetor x . A classificação será a moda das classificações das k formas de treinamento (isto é, a classificação mais freqüente).

5 Resultados Experimentais

Aplicamos o nosso método para o reconhecimento supervisionado dos gestos estáticos de mão. Usamos parte dos contornos dos gestos estáticos de mão considerados por Milios e Petrakis [10].

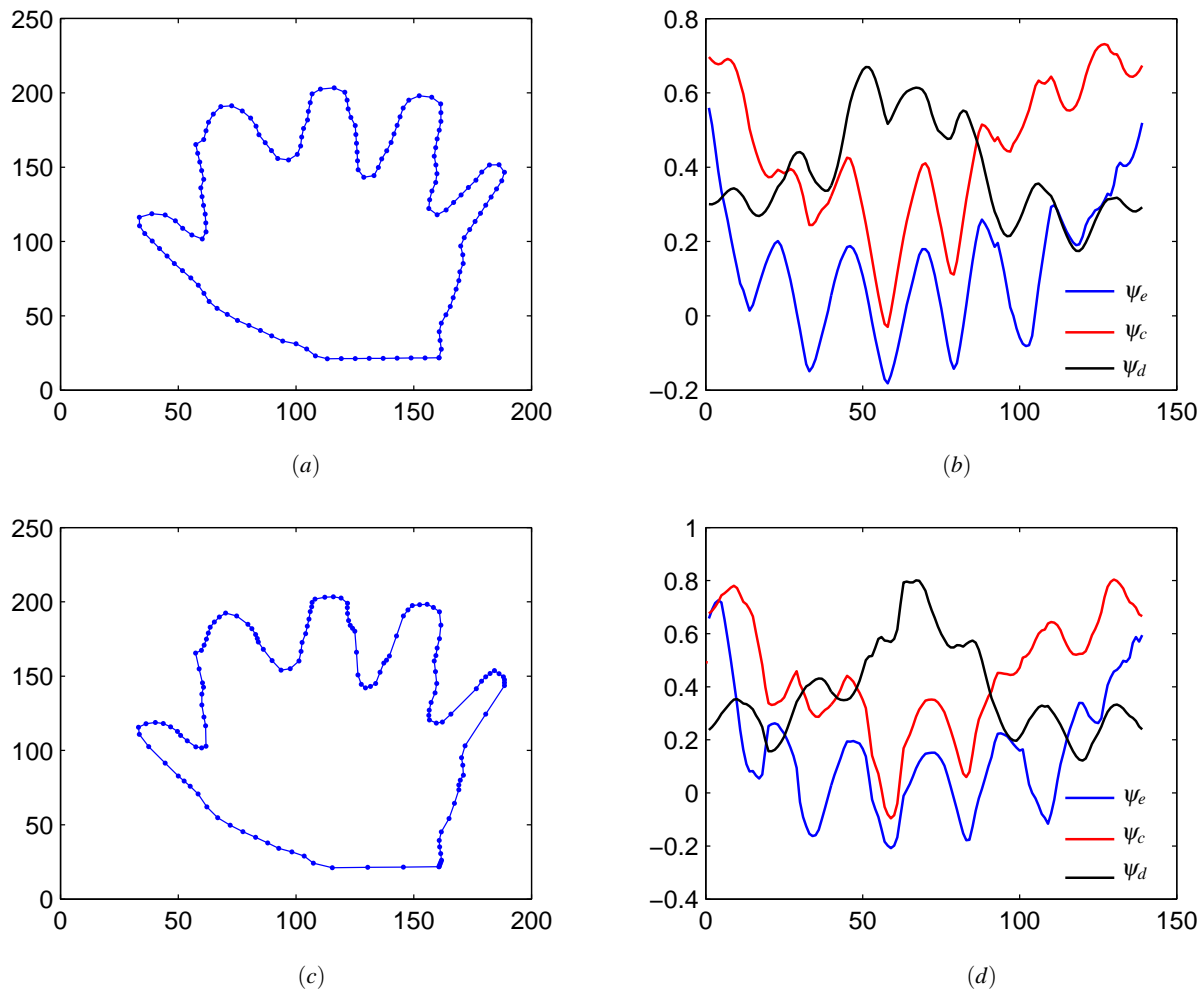


Figura 6 Vetores ψ_e , ψ_c e ψ_d de uma mesma forma com parametrizações diferentes.

Rotulamos manualmente 150 contornos fechados como exemplos de treinamento supervisionado. Eles foram divididos em seis classes, cada uma com 25 exemplos, conforme ilustrado na Figura 9. Todos os contornos foram reparametrizados usando o parâmetro comprimento de arco afim e normalizados para terem 140 pontos. A seguir, mostramos os resultados da classificação sobre formas deformadas e não-deformadas.

5.1 Formas Deformadas

Testamos a invariância afim das nossas características, especificamente para as transformações de rotação e deformação. Tomamos um contorno representativo de cada classe (que não pertence ao conjunto de treinamento) e geramos 9 versões rotacionadas, rodando o contorno original por 20° , 40° , ..., 180° . Depois, deformamos as 6×9 versões rotacionadas com as taxas de deformação $\kappa = 1, 2$ e 3 , obtendo $3 \times 6 \times 9$ formas deformadas. As Figuras 10, 11 e 12 ilus-

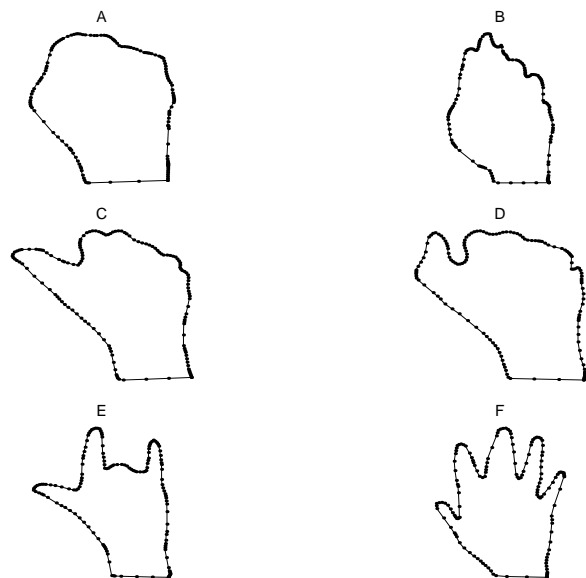


Figura 9 Gestos de mão das classes A, B, C, D, E e F.

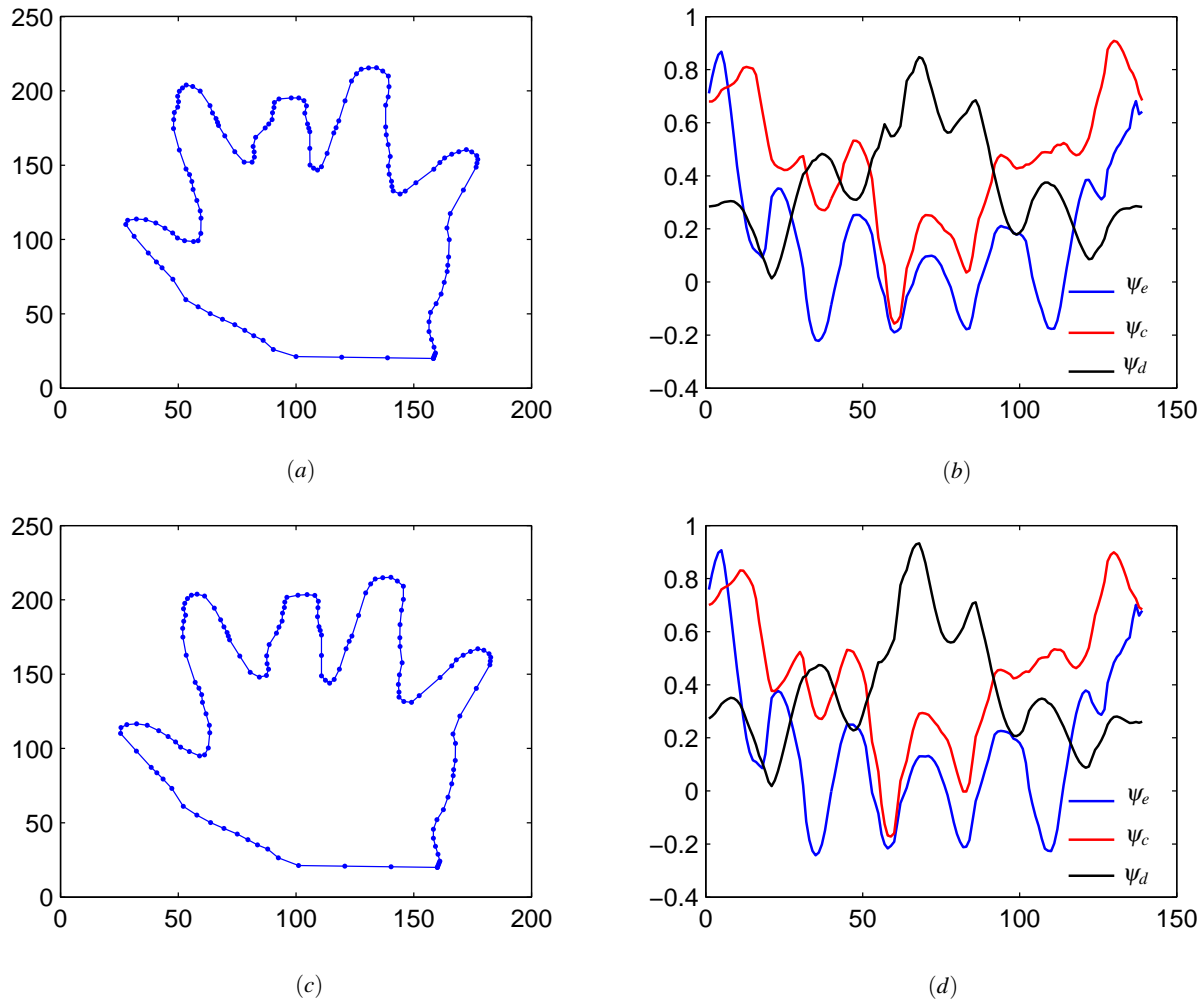


Figura 7 Vetores ψ_e , ψ_c e ψ_d de duas formas similares com parametrização pelo comprimento de arco afim.

tram a forma original e as versões deformadas com as taxas de deformação $\kappa = 1, 2$ e 3 , respectivamente.

Para a fase de reconhecimento, usamos o algoritmo dos k -vizinhos mais próximos. Testamos este algoritmo com k variando de 1 até 25. Todas as formas deformadas foram corretamente classificadas, para qualquer $k \in \{1, \dots, 25\}$. Isto mostra a robustez das características propostas sob transformação afim.

5.2 Formas Não-Deformadas

Também rotulamos manualmente 150 contornos de consulta não-deformados, com 25 contornos em cada classe. O conjunto dos contornos de consulta é disjunto do conjunto dos contornos de treinamento. A Tabela 2 resume a classificação dos gestos de mão de consulta para as formas não-deformadas, usando o algoritmo dos 5-vizinhos mais próximos. Nesta tabela, o elemento (linha i , coluna j) corresponde ao número de gestos de mão manualmente rotulados como i e classificados pelo algoritmo como j . A taxa de

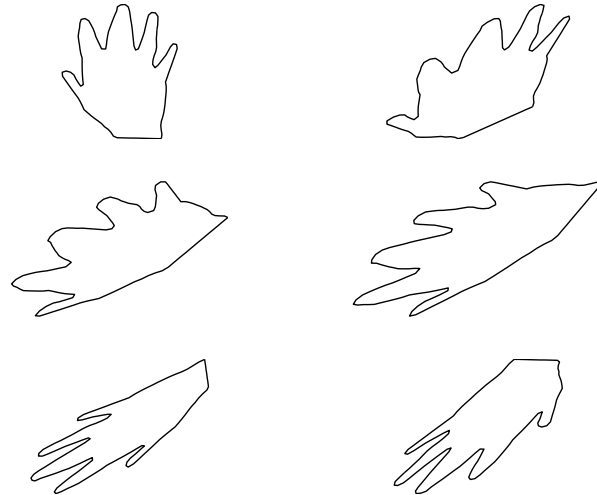


Figura 10 O gesto na parte superior esquerda pertence à classe F. Os outros são as formas deformadas com $\kappa = 1$.

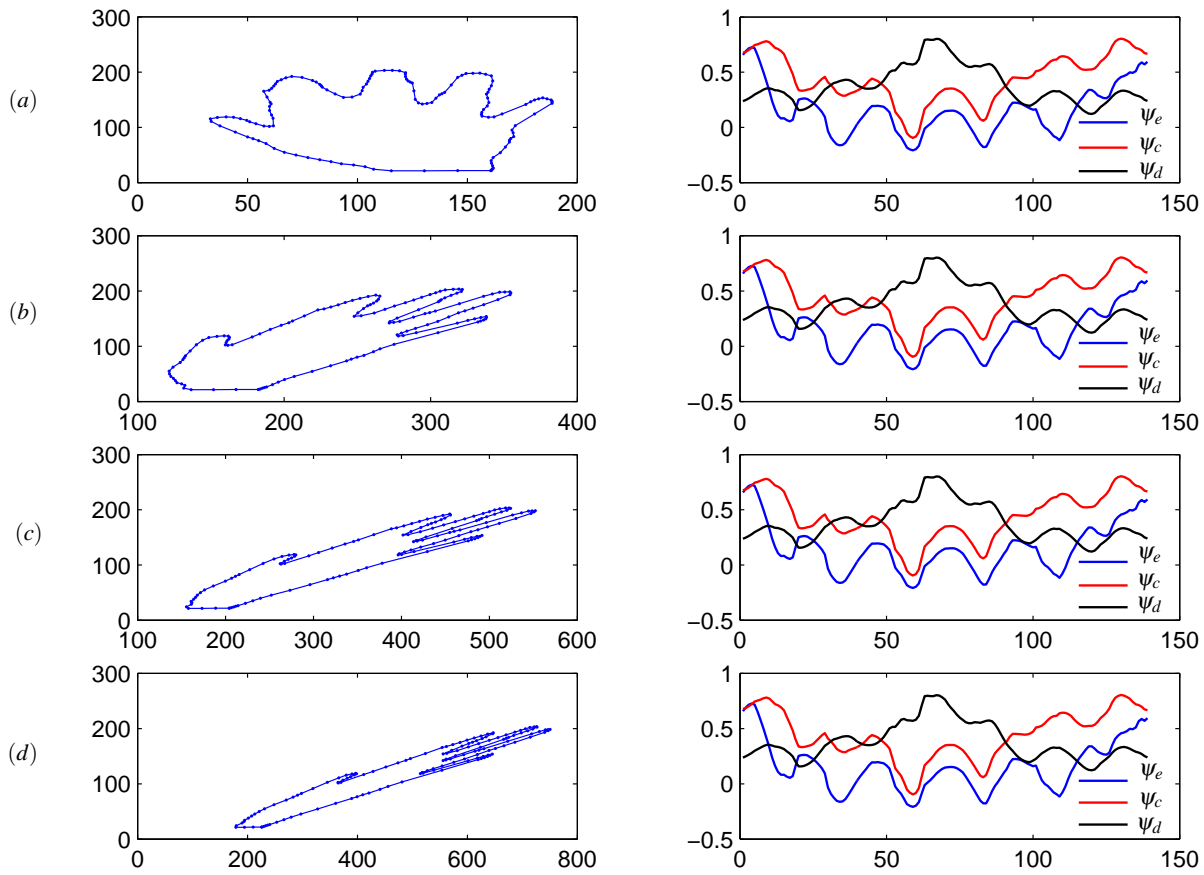


Figura 8 Vetores ψ_e , ψ_c e ψ_d sob transformação de inclinação.

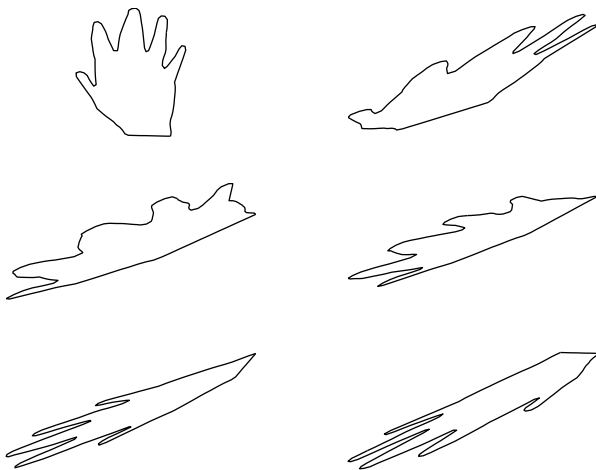


Figura 11 O gesto na parte superior esquerda pertence à classe F. Os outros são as formas deformadas com $\kappa = 2$.

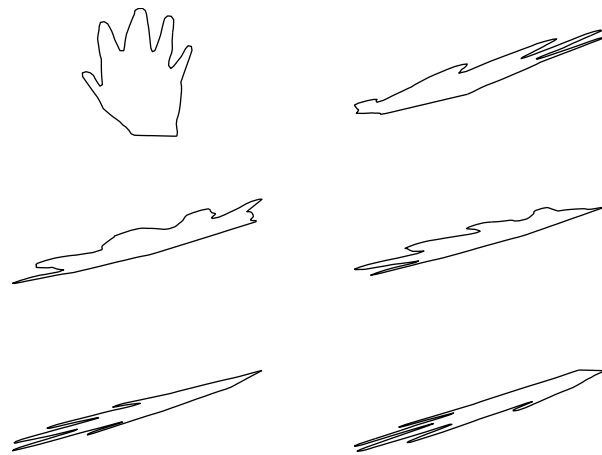


Figura 12 O gesto na parte superior esquerda pertence à classe F. Os outros são as formas deformadas com $\kappa = 3$.

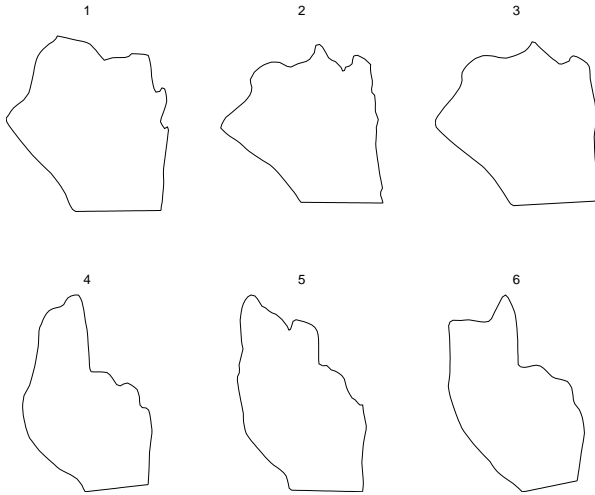
erro é muito baixa, o que mostra a aptidão das características propostas para a classificação de formas. As Figuras 13, 14 e 15 ilustram alguns contornos de consultas não-deformados e as suas classificações.

6 Conclusões

Neste capítulo, propusemos um conjunto de novas características invariante sob transformação afim. Utilizamos estas características para o reconhecimento supervisionado de formas, mas elas podem também ser usadas no reco-

Tabela 2 Classificação dos contornos de consulta.

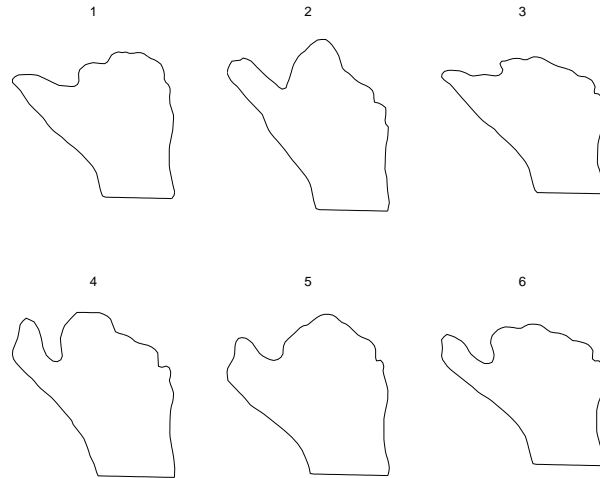
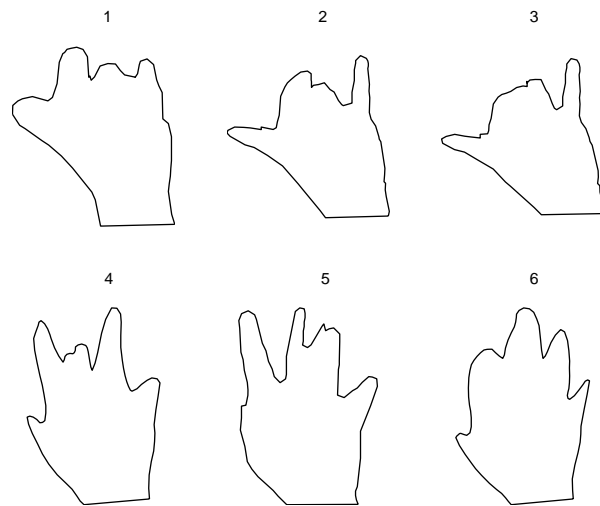
	A	B	C	D	E	F
A	25	0	0	0	0	0
B	1	24	0	0	0	0
C	0	0	25	0	0	0
D	0	1	0	24	0	0
E	0	0	0	0	25	0
F	0	0	0	1	0	24

**Figura 13** Os gestos 1, 2 e 3 pertencem à classe A e eles foram corretamente classificados pelo nosso algoritmo. Os gestos 4, 5 e 6 pertencem à classe B. Os gestos 4 e 5 foram corretamente classificados, mas o gesto 6 foi classificado como A.

nhcimento não-supervisionado de formas e na indexação e recuperação de imagens baseado em contorno. Parametrizamos as formas usando o parâmetro comprimento de arco afim, para tornar as nossas características robusto a parametrização da forma. Os resultados experimentais mostraram que as nossas características são capazes de discriminar corretamente formas severamente deformadas sob transformação afim.

Referências

1. Arbter, K., Snyder, W., H. Burkhardt, G.H.: Applications of affine-invariant fourier descriptors to recognition of 3-d objects. *IEEE Trans. Pattern Analysis and Machine Intelligence* **12**(7), 640–646 (1990)
2. Arica, N., Vural, F.T.Y.: Bas: a perceptual shape descriptor based on the beam angle statistics. *Pattern Recognition Letters* **24**(9-10), 1627–1639 (2003)
3. Attneave, F.: Some informational aspects of visual perception. *Psychological Review* **61**, 183–193 (1954)
4. Chaker, F., Bannour, M.T., Ghorbel, F.: A complete and stable set of affine-invariant fourier descriptors. In: *Proceedings of International Conference on Image Analysis and Processing*, pp. 578–581. Mantova, Italy (2003)
5. Fischler, M.A., Wolf, H.C.: Locating perceptually salient points on planar curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **16**(2), 113–129 (1994)
6. Flusser, J., Suk, T.: Pattern recognition by affine moment invariants. *Pattern Recognition* **26**(1), 167–174 (1993)
7. Fortuna, A.O.: *Técnicas Computacionais para Dinâmica dos Fluidos - Conceitos Básicos e Aplicações*. Editora da Universidade de São Paulo, São Paulo (2000)
8. Huang, Z., Cohen, F.: Affine-invariant b-spline moment for curve matching. In: *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 490–495. Seattle, WA (1994)
9. Latecki, L.J., Lakämper, R., Eckhardt, U.: Shape descriptors for non-rigid shapes with a single closed contour. In: *Proceedings of IEEE Conference Computer Vision and Pattern Recognition*, vol. 1, pp. 424–429. South Carolina, USA (2000)
10. Milios, E., Petrakis, E.G.M.: Shape retrieval based on dynamic programming. *IEEE Transactions on Image Processing* **9**(1), 141–146 (2000)
11. Mokhtarian, F.: Silhouette-based isolated object recognition through curvature-scale space. *IEEE Trans. PAMI* **17**(5), 539–544

**Figura 14** Os gestos 1, 2 e 3 pertencem à classe C e eles foram corretamente classificados pelo nosso algoritmo. Os gestos 4, 5 e 6 pertencem à classe D. Os gestos 4 e 5 foram corretamente classificados, mas o gesto 6 foi classificado como B.**Figura 15** Os gestos 1, 2 e 3 pertencem à classe E e eles foram corretamente classificados pelo nosso algoritmo. Os gestos 4, 5 e 6 pertencem à classe F. Os gestos 4 e 5 foram corretamente classificados, mas o gesto 6 foi classificado como D.

- (1995)
12. Mokhtarian, F., Abbasi, S.: Affine curvature scale space with affine length parametrisation. *Pattern Analysis & Applications* **4**, 1–8 (2001)
 13. Mokhtarian, F., Abbasi, S.: Shape similarity retrieval under affine transforms. *Pattern Recognition* **35**(1), 31–41 (2002)
 14. Mokhtarian, F., Mackworth, A.K.: A theory of multiscale, curvature-based shape representation for planar curves. *IEEE Pattern Analysis and Machine Intelligence* **14**(8), 789–805 (1992)
 15. O'Rourke, J.: *Computational Geometry in C*, 2nd edn. Cambridge University Press, New York, NY (1998)
 16. Preparata, F., Shamos, M.: *Computational Geometry: An Introduction*. Texts and Monographs in Computer Science. Springer-Verlag, New York (1985)
 17. Rattarangsi, A., Chin, R.: Scale-based detection of corners of planar curves. *IEEE Trans. PAMI* **14**(4), 430–449 (1992)
 18. Rivlin, E., Weiss, I.: Local invariants for recognition. *IEEE Trans. PAMI* **17**(3), 226–238 (1995)
 19. Sapiro, S., Tannenbaum, A.: Affine invariant scale-space. *International Journal of Computer Vision* **11**(1), 25–44 (1993)
 20. Shen, D., Wong, W., Ip, H.: Affine-invariant image retrieval by correspondence matching of shapes. *Image and Vision Computing* **17**, 489–499 (1999)
 21. Stanislaw, M.: *Description invariante et locale des formes planes - application à l'indexation d'une base d'images*. Doutorado, UNIVERSITE DE VALENCIENNES ET DU HAINAUT CAMBRESIS - ENIC/INT (1999)
 22. Zhao, A., Chen, J.: Affine curve moment invariants for shape recognition. *Pattern Recognition* **30**(6), 895–901 (1997)