Autenticação de *fingerprints*

Pedro Ribeiro Mendes Júnior, Antonio Carlos de Nazaré Júnior, David Menotti
Departamento de Computação
Universidade Federal de Ouro Preto
pedrormjunior@gmail.com, acnazare@gmail.com, menottid@gmail.com

Abstract

O problema do reconhecimento de uma pessoa por meio de impressões digitais pode ser categorizado em dois tipos: autenticação e identificação. Neste trabalho é apresentado uma abordagem por meio de MDANN *(Multi-dimensional Artificial Neural Network) que utiliza para verificar se duas impressões digitais diferentes correspondem a um mesmo dedo a partir das minúcias extraídas de cada impressão digital.*

1. Introdução

Uma impressão digital (*fingerprint*), é o desenho formado pelas papilas (elevações da pele), presentes nas polpas dos dedos das mãos. As *fingerprints* são únicas em cada indivíduo. Tal característica, chamada unicidade, as fazem serem utilizadas como forma de identificação de pessoas há séculos [4].

Os sistemas de autenticação de impressão digital*fingerprints* estão normalmente associados à identificação criminal, porém atualmente também estão sendo usados em aplicações comerciais, tais como controle de acesso e segurança em transações financeiras. Outro Tipo de aplicação que utiliza fingerprints é a confirmação de identidade de candidatos à processos seletivos e concursos públicos.

Há dois tipos de aplicações para os sistemas de reconhecimento de *fingerprints*: autenticação e identificação. Em um sistema de autenticação, a entrada do sistema é uma consulta a duas *fingerprints*, o qual verifica se ambas são pertencentes ao mesmo indivíduo. Em um sistema de identificação, a entrada é somente uma consulta a uma *fingerprint* e a saída é uma pequena lista de indivíduos que, de acordo com uma tolerância ao erro, que possivelmente possuem aquela *fingerprint*.

Uma *fingerprint* é formada por um grupo de curvas (*ridges*). As características mais comuns, chamadas de minúcias, incluem os fim de linhas, as extremidades de linha, as bifurcações e as ilhas, como ilustra a Figura 1.

Figura 1. Exemplos de Minúcias

Geralmente os algoritmos para autenticação de *fingerprints* possuem três etapas:

1. **Pré-Processamento:** É constituído pela aplicação de uma série de técnicas de processamento de imagens com o objetivo de realçar a imagem da *fingerprint*, aumentando o contraste entre as *ridges* e o fundo.

2. **Extração de Minúcias:** Extrai o conjunto de características válidas que representam a *fingerprint*.

3. **Casamento das Minúcias:** Encontra as minúcias correspondentes entre as *fingerprints* afim de verificar a semelhança entre elas.

Neste trabalho é proposto o uso de uma *MDANN* (Multidimensional Artificial Neural Network) para realizar a verificação de duas *fingerprints*.

O restante deste trabalho é organizado da seguinte forma: A Seção 2 descreve as técnicas de realce de imagens utilizadas juntamente com as técnicas de Binarização e de Afinamento. A Seção 3 explica a maneira como as minúcias foram extraídas. A Seção 4 apresenta a abordagem utilizando a *MDANN*. Por fim a conclusão e os trabalhos futuros são apresentados na Seção 5.

2. Pré-processamento

Para que haja um realce na imagem da *fingerprint*, é utilizado um algoritmo local de equalização de histograma [5].

(a) Imagem original. (b) Imagem realçada.

Figura 2. Exemplo de pré-processamento sobre a imagem original para blocos de tamanho 32×32.

A imagem é dividida em blocos de *pixels* de tamanho $h \times w$ e para cada bloco é contado o número de *pixels* de cada nível de intensidade (é calculado o histograma) e calculado o novo nível de intensidade para cada um dos 256 níveis:

$$B_i = \left(\sum_{j=0}^{i} N_j \right) \cdot \frac{intensidade\_max}{quantidade\_pixels}, i = 0, \ldots, 255 \ ,$$

onde B_i é o novo nível de intensidade no bloco, $instensidade\_max$ é 255, $quantidade\_pixels$ é igual à $h.w$ e a expressão entre parênteses representa o número de *pixels* da imagem inicial com nível de intensidade menor ou igual à i.

Nas imagens da Figura 2 é mostrado um exemplo deste processamento.

2.1. Segmentação

Neste trabalho, são realizadas duas segmentações: a primeira (descrita nesta seção) tem como objetivo localizar o "corpo" da *fingerprint* e descartar o *background*. A segunda segmentação é realizada juntamente com a extração de minúcias (Seção 3) e tem como objetivo descartar falsas minúcias que são localizadas nas bordas do "corpo" da *fingerprint* (Figura 8(c)).

Nesta primeira segmentação, é realizada uma binarização da *fingerprint* original com limiar determinado pelo método de Otsu [7]. Em seguida, no complemento da imagem binária, é realizada uma operação de fechamento para que as cristas (valor 1 na imagem de complemento) da *fingerprint* se unam entre si e a *fingerprint* se torne um único objeto sem "buracos". Na imagem resultante é aplicado uma operação de abertura para que objetos pequenos (menores que o elemento estruturante da operação) sejam excluídos.

Dessa imagem, é conseguido o menor *bounding box* que engloba o objeto (ou mais de um objeto em alguns casos). Daí, a imagem segmentada é a região do *bounding box*, correspondente à imagem original.

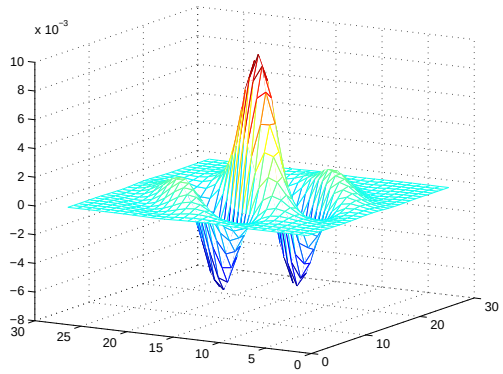


Figura 3. Representação gráfica do filtro de Gabor com os parâmetros $\theta = 3\pi/4$, $f = 1/8$, $\sigma_x = 4$ e $\sigma_y = 4$.

2.2. Filtro de Gabor

O filtro de Gabor é definido por um plano de ondas sinusoidais envolvido por uma gaussiana (Figura 3).

Uma vez que as orientações das cristas em uma mesma *fingerprint* é variável, e a aplicação do filtro de Gabor leva em consideração essa orientação (além da frequência das cristas), é necessário a construção de um novo filtro para cada região. Isso é realizado partindo a imagem em blocos e para cada bloco é calculado a orientação das cristas, frequências das mesmas, construído o filtro e aplicado.

2.2.1 Cálculo da Orientação e Frequência das Cristas

O método de cálculo da orientação e frequência das cristas descrito abaixo é realizado levando em consideração um bloco de tamanho $h \times w$.

A orientação de cada bloco pode ser calculada com [8]:

$$\Theta_B = \frac{1}{2} \text{atan} \left(\frac{\sum_{i=1}^{h} \sum_{j=1}^{w} 2 g_x(i,j) g_y(i,j)}{\sum_{i=1}^{h} \sum_{j=1}^{w} (g_x^2(i,j) - g_y^2(i,j))} \right) + \frac{\pi}{2} \ ,$$

onde g_x e g_y são as magnitudes do gradiente na direção x e y repectivamente e pode ser calculado utilizando o operador de Sobel por exemplo [5].

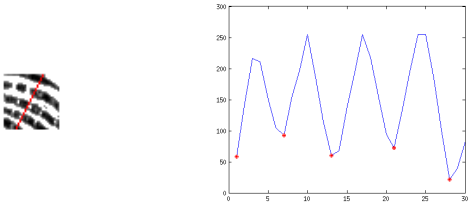
Na imagem da Figura 4 é mostrado um exemplo de aquisição das orientações com a aplicação em blocos de tamanho 16×16 e tamanho 32×32. Nestes exemplos, as imagens pré-processadas utilizadas na aquisição das orientações foram processadas em blocos do mesmo tamanho que os blocos da aquisição das orientações.

2.2.2 Frequência das Cristas

Para o cálculo da frequência das cristas em um bloco, é necessário a informação da orientação das cristas nesse bloco. É realizada uma projeção dos níveis de cinza do

(a) Blocos de tamanho 32×32 (b) Blocos de tamanho 16×16

Figura 4. Exemplos de aquisição das orientações.



(a) Reta ortogonal à orientação das cristas. (b) Projeção dos níveis de cinza.

Figura 5. Aquisição da frequência local.

bloco em uma reta através do centro ortogonal à orientação das cristas (Figura 5).

Nessa projeção é aplicado um filtro da média de tamanho 3 com o objetivo de eliminar falsos picos negativos e, em seguida, é desconsiderado o valor inicial e final do vetor de projeção. A partir dessa projeção, é calculado a distância entre picos negativos e em seguida é realizada a média das distâncias conseguidas. A média conseguida anteriormente é o comprimento de onda, e desse modo a frequência é o inverso dessa média (do comprimento de onda).

2.2.3 Filtro

A contrução do filtro de Gabor é dada pela seguinte equação:

$$G(x,y) = \frac{1}{2\pi\sigma_x\sigma_y}\exp\left(-\frac{1}{2}\left(\frac{x_\theta^2}{\sigma_x^2}+\frac{y_\theta^2}{\sigma_y^2}\right)\right)\cos\left(2\pi x_\theta f\right) ,$$

onde

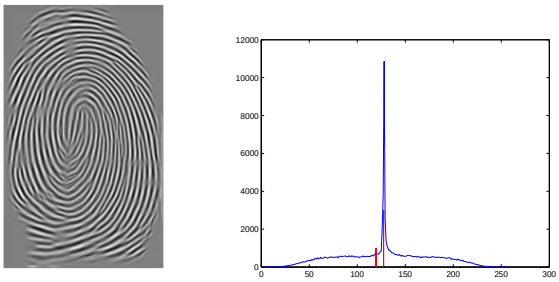
$$x_\theta = x\cos(\theta) + y\sin(\theta) \text{ e}$$
$$y_\theta = -x\sin(\theta) + y\cos(\theta) .$$

Aqui, σ_x e σ_y são escolhidos em função da frequência [6]:

$$\sigma_x = \frac{k_x}{f}, \sigma_y = \frac{k_y}{f} ,$$

e o tamanho do filtro em função de σ_x e σ_y [6]:

$$w_x = 4.5\sigma_x, w_y = 4.5\sigma_y .$$



(a) Após o filtro de Gabor (b) Histograma após o filtro de Gabor

Figura 6. Detecção do limiar

2.3. Binarização

Em uma imagem de retorno do filtro de Gabor, há duas classes de valores que deseja-se diferenciar: o mais escuro referente às cristas e o mais claro referente aos vales (entre cristas) como visto na Figura 6(a). Nessa imagem há uma grande concentração de valores que estão entre os valores dessas duas classes citadas, que são referentes ao *background* e outras regiões de indecisão como regiões referentes às singularidades (deltas e loops por exemplo) ou falhas no desenho da *fingerprint* (Figura 6(b)).

Desse modo, é calculado a média (maior reta vertical da Figura 6(b)) e variância, dado que a imagem após a aplicação do filtro de Gabor está normalizada entre 0 e 1. O limiar T é determinado simplesmente calculando-se

$$T = \mu - \sigma^2 ,$$

onde μ é a média e σ^2 é a variância das intensidades de *pixels* da imagem após a aplicação do filtro de Gabor. No histograma da Figura 6(b), a menor reta vertical representa esse limiar.

A subtração da variância na média é realizada porque, uma vez que se quer o *background* da imagem (*pixels* cuja intensidade está entre $\mu-\sigma^2$ e $\mu+\sigma^2$) branco no momento da binarização (para que fique semelhante à imagem original), o limiar é o mais próximo à média de modo à deixar o *background* branco.

2.4. Afinamento

Após a binarização (Figura 7(a)), é realizado um afinamento nas cristas da imagem para que na etapa de extração de minúcias se possa detectar as terminações e bifurcações. O afinamento (Figura 7(b)) é realizado utilizando operações morfológicas [7].

3. Extração de Minúcias

Após o realce da imagem da *fingerprint* a mesma é utilizada para a extração dos pontos de minúcias. Existe várias características que podem ser utilizadas para autenticar a

(a) Binarização. (b) Afinamento.

Figura 7. Binarização e afinamento.

fingerprint, mas a maioria das minúcias se restringem à apenas dois tipos: As bifurcações e as extremidades de linha, aqui chamada de terminações.

Uma terminação é o ponto onde uma linha (*ridge*) termina e as bifurcações são os pontos onde o *ridge* se divide de um caminho simples para uma junção em 'Y';

3.1. Extração de Todas as Minúcias

Os pontos de terminação e bifurcação são extraídos da imagem afinada (Figura 7) da *fingerprint* com auxílio do conceito de *Condition Number* (C_N) [1]. A equação a seguir calcula o *condition number* para um pixel pertencente ao ridge da *fingerprint*.

$$C_N = \frac{\sum_{k=1}^{8} |\Gamma(k+1) - \Gamma(k)|}{2} \text{ onde } \Gamma(9) = \Gamma(1).$$

onde

$$\Gamma(p) = \begin{cases} 1 \text{ se } p \text{ pertence ao } \textit{ridge} \\ 0 \text{ caso contrario} \end{cases} \quad (1)$$

e k representa os oito vizinhos de p ordenados na direção horária.

Se $C_N(p)$ for igual 1, p será um ponto de terminação. Para p ser uma bifurcação, $C_N(p)$ deverá ser igual 3. Todo os outros valores de C_N são ignorados. Ao final têm-se uma matriz bidimensional ($MIN_{i,j}$ com as mesmas dimensões i e j da imagem afinada contendo os valores 1 e 3 para os pontos minúcias e os valores restantes iguais à 0.

A Figura 8(a) apresenta todas as minúcias extraídas da *fingerprint* da Figura 7.

3.2. Filtragem de Minúcias Espúrias

Após a extração das minúcias, as mesmas passaram por um filtro afim de detectar minúcias espúrias e removê-las. Estas minúcias são frequentemente encontradas nas amostras de *fingerprint* devido à presença de ruído nas outras etapas do processamento da imagem e diminuem a precisão e desempenho da autenticação das *fingerprints*.

A primeira parte desta filtragem consiste em excluir todas as minúcias pertencentes à extremidade da *fingerprint*. Para tanto é criada uma mascara binária (MB) a partir da imagem afinada realizando-se operações morfológicas na mesma [2], tal máscara é apresentada na Figura 8(b) Serão consideradas minúcias verdadeiras apenas aquelas que intercederem com a máscara. Esta operação pode ser expressa pela seguinte equação:

$$Min'(i,j) = Min(i,j) \times MB(i,j),$$

onde $Min'(i,j)$ é a nova matriz de pontos de minúcias. A Figura 8(c) apresenta o resultado desta operação.

A segunda parte da filtragem de minúcias espúrias consiste na análise do fluxo das *ridges*, bem como a distância e a conectividade das minúcias. O algoritmo utilizado é descrito em [2].

O resultado final da filtragem de minúcias espúrias é apresentado na Figura 8(d).

4. Casamento

O método estudado neste trabalho para realizar o casamento entre *fingerprints* utiliza redes neuroniais artificiais multidimensional (MDANN - *Multi-Dimensional Artificial Neural Network*) e foi proposto por [3].

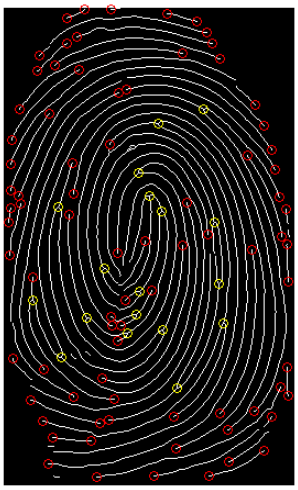
Neste método, são utilizadas as etapas descritas anteriormente, isto é, segmentação, aplicação do filtro de Gabor, binarização, afinamento e extração de minúcias do tipo terminação e bifurcação.

4.1. MDANN

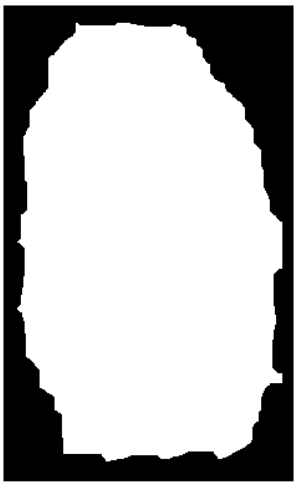
Dado que a utilização da imagem de minúcias (Figura 8(d)) para o treinamento em uma MDANN se torna inviável, é criado uma matriz de tamanho 15×15 para cada imagem de minúcias. Essa matriz (matriz de entrada para a MDANN) é construída da seguinte maneira para uma dada imagem de minúcias: A imagem é dividida em 15 por 15 blocos. Para cada bloco, é realizado um somatório dos *pixels* abarcados por ele, sabendo-se que os valores de *pixel* da imagem de minúcias são 0 (não há minúcias), 1 (tipo terminação) ou 3 (tipo bifurcação). O resultado de cada somatório é um elemento da matriz.

Na implementação realizada de acordo com as descrições dadas por [3], não havia convergência do método. Com a análise da rede de [3], foi observado que no cálculo do valor de saída da rede havia um erro, acarretando a divergência no aprendizado.

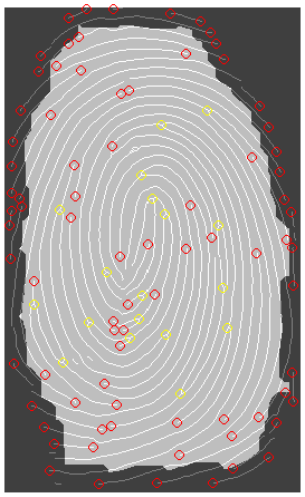
Neste trabalho, foi realizada uma alteração na função de saída, o que fez com que a rede não divergisse. Apesar disso, o menor erro quadrático médio (MSE - *Mean*



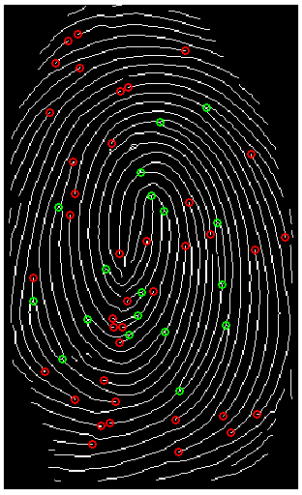
(a) Todas as minúcias extraídas



(b) Mascára Binária para remoção de minúcias nas extremidades da *fingerprints*



(c) Máscara aplicada à matriz de pontos de minúcias



(d) Imagem de minúcias após a filtragem de minúcias espúrias

Figura 8. Exemplo da filtragem de minúcias espúrias

Square Error) conseguido fica em torno de $0,3$ (Figura 9). A alteração realizada no cálculo da saída foi somente a multiplicação por -1 no argumento da exponencial, ou seja, ao invés de calcular o valor dado pela Equação 2, calcula-se o valor dado pela Equação 3.

$$output = A_w(1) + \sum_{j=2}^{n} A_w(j) \times \Omega_1 \quad (2)$$

$$output = A_w(1) + \sum_{j=2}^{n} A_w(j) \times \Omega_2 \quad (3)$$

Onde:

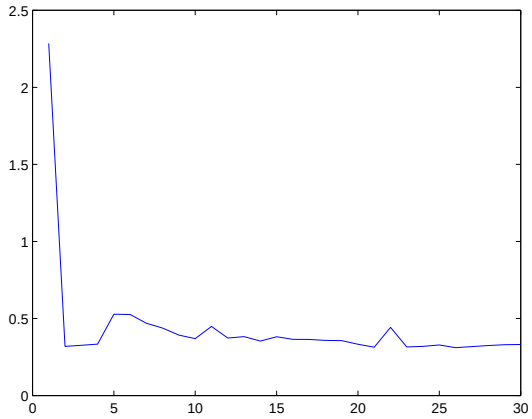


Figura 9. Número de épocas por MSE para $\alpha = 0,5$.

$$\Omega_1 = \left(\frac{2}{1 + \exp((S_{j1} \times I(i) \times S_{j2}) + A_b(j))} - 1 \right)$$

$$\Omega_2 = \left(\frac{2}{1 + \exp(-(S_{j1} \times I(i) \times S_{j2}) + A_b(j))} - 1 \right)$$

5. Conclusão e Trabalhos Futuros

O principal estendimento deste trabalho está em uma análise mais aprofundada do método MDANN, de modo à descobrir o motivo pelo qual o método apresenta tão alta taxa de erro na aprendizagem (MSE). Além disso, dada a etapa de pré-processamento descrita neste trabalho (Seção 2), objetiva-se estudar outros métodos de casamento de *fingerprints* que a utiliza e destacar seus prós e contras, assim como aperfeiçoar tal pré-processamento.

Referências

[1] J. C. Amengual, J. C. Prez, F. Prat, and J. M. Vilar. Realtime minutiae extraction in fingerprint images. In *Proceedings of the 6th International Conference on image processing and its application*, pages 871–875, 1997.

[2] S. Kim, D. Lee, and J. Kim. Algorithm for detection and elimination of false minutiae in fingerprint images. In J. Bigun and F. Smeraldi, editors, *Audio- and Video-Based Biometric Person Authentication*, volume 2091 of *Lecture Notes in Computer Science*, pages 235–240. Springer Berlin / Heidelberg, 2001.

[3] R. Kumar and B. R. D. Vikram. Fingerprint matching using multi-dimensional ann. *Engineering Applications of Artificial Intelligence*, 23(2):222–228, 2010.

[4] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar. *Handbook of Fingerprint Recognition*. Springer, 2 edition, 2009.

[5] R. Miron and T. Letia. Fuzzy logic decision in partial fingerprint recognition. In *IEEE International Conference on Automation, Quality and Testing, Robotics*, volume 3, pages 1–6, 2010.

[6] R. Thai. Fingerprint image enhancement and minutiae extractions. Technical report, 2003.

[7] The MathWorks. Image processing toolbox user's guide, 2010.

[8] Y. Wang, J. Hu, and F. Han. Enhanced gradient-based algorithm for the estimation of fingerprint orientation fields. *Applied Mathematics and Computations*, 185:823–833, 2007.