Estudo Comparativo de Abordagens Bioinspiradas para a Verificação de Impressões Digitais

Wesley Gomes de Brito Soares[1], **Matheus Giovanni Pires**[2]

[1]Engenharia de Computação – Universidade Estadual de Feira de Santana (UEFS)
Feira de Santana – BA – Brasil

[2]Departamento de Exatas – Universidade Estadual de Feira de Santana (UEFS)

wesley.gbs@gmail.com, mgpires@ecomp.uefs.br

Abstract. *Personal passwords are highly used in several real applications, such as, e-mail and credit cards. But, increasing password theft have brought a huge problem to both users and companies, one way to try to solve this issue is using biometric data. This article presents a comparative study between two bioinspired approaches, Ant Colony System and Genetic Algorithms, applied to fingerprint verification problem. The results are shown in ROC curves.*

Resumo. *Senhas pessoais são amplamente utilizada em vários tipos de aplicações reais, tais como, acesso ao email e no uso de cartões de débito ou crédito. Porém, o aumento do furto de tais senhas tem sido um grande problema para os usuários e empresas, e uma das formas para se tentar solucionar este problema é o uso de informações biométricas. Este artigo apresenta um estudo comparativo entre duas abordagens bioinspiradas, Colônia de Formigas e Algoritmos Genéticos, aplicadas na resolução do problema da verificação de impressões digitais. Os resultados obtidos são avaliadas através de curvas ROC.*

1. Introdução

A aplicação de senhas pessoais é a forma mais utilizada em sistemas que requerem algum nível de segurança para a identificação de usuários. Um uso prático bem conhecido é a utilização de caixas eletrônicos dos bancos para efetuar saques ou consultas de saldo de contas correntes, por exemplo.

Um outro exemplo bem conhecido da utilização de senhas pessoais é o cartão de crédito. Com o aumento das compras pela Internet, o furto das senhas pessoais em sites de comércio eletrônico tornou-se muito comum. No 1º trimestre de 2016 os brasileiros movimentaram R$ 269 bilhões com cartões de crédito e débito, crescimento de 7,2% em relação aos três primeiros meses de 2015. Os cartões de crédito registraram R$ 165 bilhões (alta de 3,8%), e os cartões de débito, R$ 104 bilhões (alta de 13%) [Associação Brasileira das Empresas de Cartão de Crédito e Serviços 2016], tornando este tipo de golpe cada vez mais atraente.

Para contornar esse e outros tipos de problemas, como identificação para acesso a locais restritos, foram desenvolvidas técnicas para a identificação automática das pessoas por meio de suas características físicas ou comportamentais, ou seja, através da biometria [Jain et al. 2002]. O uso de biometria garante maior confiabilidade em relação ao uso

de senhas, pois duas pessoas não possuem características idênticas e é quase impossível fraudar esse tipo de informação.

Este trabalho propõe um estudo comparativo entre duas abordagens, uma baseada em Colônia de Formigas e outra em Algoritmos Genéticos, para a resolução do problema da verificação de impressões digitais. A partir de uma base de dados de impressões digitais previamente cadastrada, o sistema deverá verificar se existe uma impressão digital compatível com a impressão digital do usuário que se deseja identificar.

Este artigo está organizado da seguinte maneira: Na seção 2 são apresentados conceitos referentes à Biometria, com foco na impressão digital, e Computação Bioinspirada, mais precisamente, sobre Colônia de Formigas e Algoritmos Genéticos. Na seção 3 é descrita a utilização da abordagem baseada em Colônia de Formigas e Algoritmos Genéticos para a resolução do problema da verificação de impressões digitais. Na seção 4, são descritos os testes realizados e os resultados obtidos. Por fim, na seção 5, as conclusões são apresentadas.

2. Fundamentação Teórica

2.1. Biometria

A Biometria se refere a identificação de pessoas baseadas em características físicas ou comportamentais, tais como, impressão digital, retina, íris, geometria da mão, face, voz, assinatura, dentre outros. A Biometria é mais confiável que os métodos tradicionais de identificação, pois essas técnicas são propensas a fraudes, como por exemplo, os cartões de crédito ou débito podem ser roubados e as senhas descobertas. Por outro lado, as características biológicas não podem ser esquecidas, compartilhadas ou extraviadas, requerendo ainda que a pessoa esteja presente para fornecer sua medida biométrica no instante da autenticação [Jain et al. 2002]. Para que uma característica física ou comportamental possa ser utilizada para a identificação de um indivíduo é necessário que ela atenda à quatro requisitos [Jardini 2007] [Pain et al. 2004]: Universalidade, unicidade, permanência e coletabilidade. Para o desenvolvimento de um sistema biométrico, a característica deve, ainda, atender a critérios de desempenho, aceitabilidade e evasão.

Um sistema biométrico é um sistema de reconhecimento de padrões que opera sobre uma informação biométrica de um indivíduo, extraindo um conjunto de características desta informação e comparando-o com um segundo conjunto. A depender da aplicação, um sistema pode ser definido como de *verificação* ou *identificação* [Pain et al. 2004]. Em um sistema de verificação, a entrada é uma consulta a uma informação biométrica associada com uma identidade (ID). Neste caso, o sistema verifica se a entrada é consistente ou não com a informação biométrica armazenada. Este tipo de sistema é ilustrado na Figura 1(a). Em um sistema de identificação, a entrada do sistema é somente uma consulta a uma informação biométrica (ver Figura 1(b)). Neste caso, o sistema tenta responder a seguinte questão: Existe alguma informação biométrica na base de dados que se pareça com o dado consultado? A resposta é uma pequena lista contendo os modelos com maior grau de similaridade com a entrada.

2.2. Impressão digital

A impressão digital de um indivíduo é única e permanece inalterada ao longo de sua vida. Ela é formada por cristas, as quais são definidas como segmentos de curva. Além

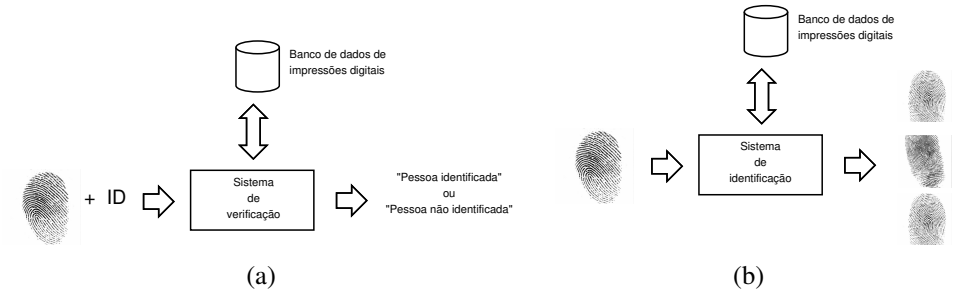


Figura 1. Sistema de verificação (a) e identificação (b) de impressões digitais

das cristas, há também os vales, que são as regiões entre duas cristas [Thai 2003]. Estas curvas apresentam características denominadas minúcias e possuem as mais variadas formas: ponto-final, bifurcação, fragmento de linha, buraco, triângulo e gancho (Figura 2) [Jain et al. 1997, Thai 2003]. Entretanto, os sistemas de identificação de impressões digitais costumam utilizar pontos-finais e bifurcações, já que as demais tendem a serem introduzidas nas imagens devido a ruídos, criando falsas minúcias.

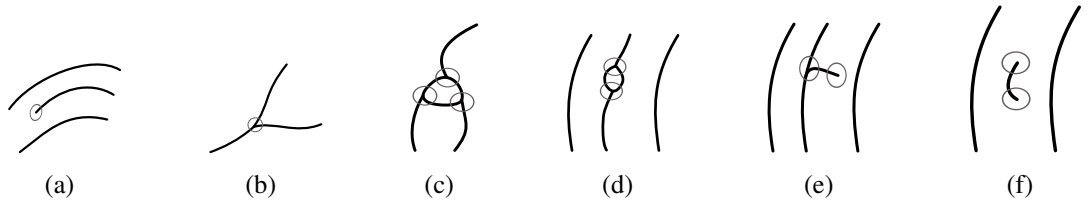


Figura 2. Exemplo de minúcias: ponto-final (a), bifurcação (b), triângulo (c), buraco (d), gancho (e) e fragmento de linha (f).

O reconhecimento de impressões digitais é um dos métodos mais usados dentre as técnicas de reconhecimento biométrico. Não é empregado apenas na aplicação das leis (identificação de suspeitos), mas também em aplicações comerciais que exigem um certo grau de segurança, como por exemplo o controle de acesso à locais restritos e transações financeiras.

2.3. Otimização Contínua usando Colônia de Formigas

A otimização por Colônia de Formigas é uma meta-heurística para a solução de problemas combinatórios baseados no comportamento forrageiro das formigas na busca de alimentos [Dorigo et al. 1999]. Um algoritmo baseado em Colônia de Formigas para a solução de problemas de otimização em domínios contínuos, denominado *ACO for continuous domains* ($\text{ACO}_{\mathbb{R}}$) foi proposto por [Socha and Dorigo 2008]. A ideia básica do $\text{ACO}_{\mathbb{R}}$ é que a construção das soluções é feita de forma incremental pelas formigas, guiadas pelo feromônio e por uma Função de Densidade de Probabilidade (FDP).

No $\text{ACO}_{\mathbb{R}}$ a informação do feromônio é armazenada em uma estrutura de dados chamada **arquivo de solução**, conforme ilustrado na figura 3. No arquivo de solução é armazenada uma quantidade k de soluções. Para cada solução s_l de um problema n-dimensional, o $\text{ACO}_{\mathbb{R}}$ armazena neste arquivo os valores de suas n variáveis e o valor de sua função objetivo $f(s_l)$, a qual é responsável por medir a qualidade da solução. A i-ésima variável da l-ésima solução é denotada por s_l^i.

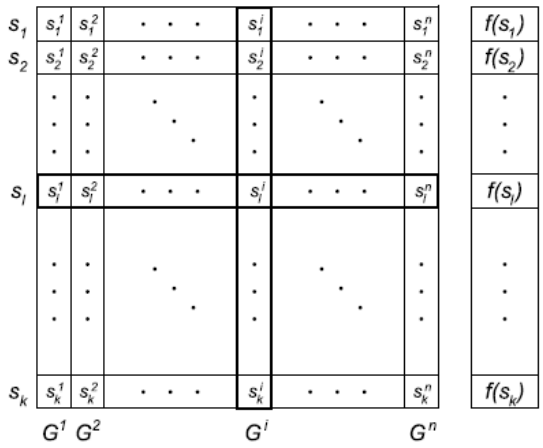


Figura 3. Representação do arquivo de solução [Socha and Dorigo 2008].

A execução do $ACO_{\mathbb{R}}$ é descrito no algoritmo 1. O processo inicia-se com a definição dos parâmetros, como por exemplo, o número de formigas, e com a criação do arquivo de solução. Em seguida, inicia-se a etapa da construção das soluções, guiadas pelo feromônio e por uma FDP. A quantidade de soluções que serão criadas é igual ao número de formigas definidas. Por fim, os valores dos feromônios são atualizados. Este processo se repete até que um critério de parada seja atingido. A melhor solução para o problema será a que tiver a melhor função objetivo.

Algoritmo 1: Pseudocódigo do $ACO_{\mathbb{R}}$ [Socha and Dorigo 2008].

```
1 Inicia os parâmetros;
2 enquanto critério de parada não atingido faça
3 |   ControiSolução();
4 |   AtualizaFeromônio();
5 |   BuscaLocal() {opcional};
```

Maiores detalhes sobre o algoritmo $ACO_{\mathbb{R}}$ podem ser encontrados em [Socha and Dorigo 2008].

2.4. Otimização usando Algoritmos Genéticos

Algoritmos Genéticos (AG) são métodos de otimização e busca baseados nos mecanismos da seleção natural introduzidos por [Holland 1975]. Devido a grande capacidade de explorar espaços de busca grandes e irregulares, os AG têm sido bastante utilizados para estes tipos de problemas. Um Algoritmo Genético simples possui uma estrutura conforme o pseudocódigo ilustrado no algoritmo 2 [Michalewicz 1996].

Durante a iteração t, o algoritmo genético mantém uma população de soluções candidatas (cromossomos). A primeira população, geralmente, é gerada de forma aleatória. Cada solução é avaliada para medir sua aptidão (ou *fitness*), ou seja, a qualidade da solução do problema representada por este cromossomo. Então, uma nova população (iteração $t+1$) é formada pela seleção favorável aos indivíduos mais aptos. Alguns

Algoritmo 2: Pseudocódigo de um algoritmo genético simples.

```
1 t ← 0;
2 Inicializa P(t);
3 Avalia P(t);
4 enquanto critério de parada não atingido faça
5     t ← t + 1;
6     Seleciona P(t) de P(t − 1);
7     Altera P(t);
8     Avalia P(t);
```

membros desta nova população sofrerão alterações devido à ação dos operadores genéticos de cruzamento (*crosssover*) e mutação (*mutation*), enquanto outros permanecerão intactos. O cruzamento combina as características de dois cromossomos pais para formar dois cromossomos filhos. O objetivo da aplicação do cruzamento é trocar informações entre soluções em potencial, objetivando uma melhor exploração de uma determinada região do espaço de busca. Por outro lado, a mutação altera aleatoriamente um ou mais genes de um cromossomo selecionado, com o intuito de introduzir informação extra para a população. Em outras palavras, a mutação contribui com a diversidade da população, proporcionando a exploração de novos locais do espaço de busca e, consequentemente, evitando os máximos ou mínimos locais.

3. Metodologia

O sistema para verificação de impressões digitais desenvolvido neste projeto é composto pelos seguintes módulos: pré-processamento das imagens, extração de minúcias, banco de dados e verificação das impressões digitais. A figura 4 mostra como estes módulos estão organizados.

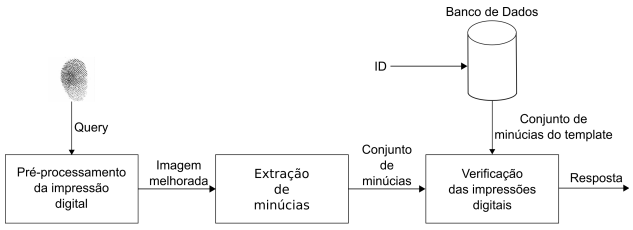


Figura 4. Organização de um sistema de verificação.

De acordo com a figura 4, a entrada do sistema é dada por duas informações: a impressão digital de um usuário (*query*) e um identificador (ID). O primeiro passo do processo de verificação é o pré-processamento da impressão digital do usuário, que tem por objetivo diminuir as imperfeições da imagem e destacar as informações relevantes, neste caso, as cristas e vales. Em seguida, são extraídas as minúcias da impressão digital. De posse do conjuntos de minúcias do usuário e do conjunto de minúcias do ID (armazenado no banco de dados), a comparação entre as impressões digitais é realizado por um algoritmo de verificação.

3.1. Pré-processamento das impressões digitais

Para assegurar que o desempenho de um sistema de verificação não seja comprometido devido a ruídos nas imagens, é necessário realizar um pré-processamento sobre as mesmas, com o objetivo de realçar as características que serão utilizadas pelo sistema.

A metodologia aplicada neste trabalho é a mesma apresentada por [Thai 2003], na qual foram utilizadas as seguintes técnicas: segmentação, normalização, estimativa de orientação, estimativa da frequência das cristas, filtro de Gabor, binarização e, por fim, afinamento.

Neste trabalho foi utilizado os algoritmos desenvolvidos por Peter Kovesi, disponíveis em http://www.peterkovesi.com/matlabfns/index.html#fingerprints, para a realização da etapa de pré-processamento das impressões digitais.

3.2. Extração de minúcias

O módulo de extração de minúcias é responsável por detectar as minúcias das imagens. Neste trabalho foram extraídas apenas dois tipos de minúcias para a realização da verificação das impressões digitais, pontos finais e bifurcações.

A extração das minúcias é baseada no algoritmo desenvolvido por [Thai 2003], chamado *Crossing Number* (CN). Após este processo, um *pixel* será classificado como ponto-final, se CN for igual a um, ou bifurcação, se for igual a três. Para cada minúcia extraída são armazenadas as seguintes informações: coordenadas x e y, ângulo de orientação da crista em que a minúcia está (obtido no passo anterior) e o tipo de minúcia (bifurcação ou ponto final).

A implementação desta etapa utilizou o código de Vahid K. Alilou, disponível em http://www.mathworks.com/matlabcentral/fileexchange/44369-fingerprint-matching–a-simple-approach.

3.3. Verificação de impressões digitais

O problema de verificação biométrica, em especial a impressão digital, pode ser formulado da seguinte forma. Sejam T e Q a representação do *template* e *query*, respectivamente, de forma que cada um deles é um vetor de minúcias: $T = \{m_1^T, m_2^T, \ldots, m_n^T\}$ e $Q = \{m_1^Q, m_2^Q, \ldots, m_p^Q\}$, onde n e p é a quantidade de minúcias em T e Q, respectivamente. Cada minúcia é descrita pela 4-upla $m = \{x, y, \theta, t\}$, onde x e y são as coordenadas espaciais, θ é o ângulo e t é o tipo da minúcia, podendo ser ponto-final ou bifurcação. O *template* representa o modelo de uma impressão digital de uma pessoa e a *query* é a entrada do sistema de verificação. A verificação de duas impressões digitais consiste da aplicação de duas etapas: (1) alinhamento e (2) comparação das minúcias. O alinhamento de duas impressões digitais é indispensável, já que permite maximizar o número de minúcias correspondentes na etapa seguinte. Este processo se dá através de reposicionamento das minúcias da *query*, usando uma função de escala (s), rotação (θ) e translação (Δx e Δy), dada pela equação 1 [Tan and Bhanu 2006][Jain and Maltoni 2009]:

$$
\begin{aligned}
&map(m^Q = \{x^Q, y^Q, \theta^Q, t^Q\}) = m^{Q'} = \{x^{Q'}, y^{Q'}, \theta^Q + \theta,, t^Q\}, \text{ onde} \\
&\begin{bmatrix} x^{Q'} \\ y^{Q'} \end{bmatrix} = s \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x^Q \\ y^Q \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}
\end{aligned} \tag{1}
$$

Em seguida, é realizada a comparação, onde duas minúcias m_i^T e $m_j^{Q'}$ são consideradas correspondentes se a diferença espacial (Δe) (equação 2) e diferença angular ($\Delta\theta$) (equação 3) são inferiores a um limiar D_0 e θ_0, respectivamente [Jain and Maltoni 2009]:

$$\Delta e(m_i^T, m_j^{Q'}) = \sqrt{(x_i^T - x_j^{Q'})^2 + (y_i^T - y_j^{Q'})^2} \leq D_0, \text{ e} \tag{2}$$

$$\Delta\theta(m_i^T, m_j^{Q'}) = min(|\theta_i^T - \theta_j^{Q'}|, 360^o - |\theta_i^T - \theta_j^{Q'}|) \leq \theta_0 \tag{3}$$

Este cálculo é feito minúcia a minúcia, de forma que uma minúcia $m_j^{Q'}$ pode ter apenas uma minúcia m_i^T correspondente. Após o processamento, obtém-se um conjunto **C** de minúcias correspondentes, com o qual é calculado o grau de similaridade, ou *score*, entre T e Q, conforme a equação 4 [Jain and Maltoni 2009]. Se o *score* for maior que um limiar L, as impressões digitais pertencem ao mesmo indivíduo.

$$score = \frac{|C|}{\frac{n+p}{2}} \tag{4}$$

O problema da verificação entre duas impressões digitais consiste em alinhar as duas impressões, a fim de possibilitar a comparação das mesmas. Para que este alinhamento contribua com resultados satisfatórios, é necessário encontrar bons valores para as variáveis da equação 1. Este processo consiste em um problema de otimização, que neste trabalho será resolvido por Colônia de Formigas e Algoritmos Genéticos.

3.4. Modelagem do problema usando Colônia de Formigas

O arquivo de solução tem seus componentes dispostos em quatro colunas, contendo s, θ, Δx e Δy. Para inicializar esta estrutura são gerados valores iniciais aleatoriamente e igualmente espaçados, impedindo que duas ou mais soluções estejam associadas a uma mesma região. A função objetivo adotada é o grau de similaridade entre o *template* e a *query*, portanto, utilizou-se a equação 4.

3.5. Modelagem do problema usando Algoritmos Genéticos

A modelagem do Algoritmo Genético é bastante similar à do $ACO_{\mathbb{R}}$. A codificação cromossômica baseia-se no uso dos mesmos quatro componentes, com uso da representação real das variáveis, assim a manipulação da informação torna-se mais fácil. A população inicial é grande, e a cada geração dois novos filhos são obtidos através de um operador de cruzamento aritmético para representação real. O operador de mutação é uma adaptação do *bit-flip* para a representação real, onde é aleatoriamente adicionado ou subtraído um valor ao gene escolhido. A função de *fitness* é exatamente igual a função objetivo definida pela equação 4.

4. Experimentos e Resultados

Todos os experimentos foram executados sobre a base de dados DB1, subconjunto B, da *Fingerprint Verification Competition* (FVC), que ocorreu em 2002 [BioLab - University of Bologna 2002]. Esta base de dados caracteriza-se por ter imagens

adquiridas pelo sensor óptico *TouchView II*, cada uma delas com tamanho de 388x374 (142 Kpixels) e resolução de 500 dpi. Ela conta com dois conjuntos de impressões digitais denominados **A** e **B**. O primeiro possui 110 impressões digitais diferentes, cada uma com 8 amostras. O segundo é um subconjunto do primeiro, contendo 10 impressões digitais diferentes e 8 amostras de cada.

A forma de avaliação adotada neste trabalho é o mesmo utilizado na FVC, a qual é dividida em verificação de genuínos e de impostores. Na verificação de genuínos, cada amostra do conjunto é comparada às amostras remanescentes do mesmo dedo. Já na verificação de impostores, a primeira amostra de cada dedo é comparada à primeira amostra dos outros dedos. Dessa forma, haverão 280 comparações para os testes de genuínos e 45 para os impostores. Os resultados obtidos foram apresentados por curvas ROC [Fawcett 2006]. As curvas ROC geradas expressam as taxas de verdadeiro positivo por falso positivo, possibilitando a análise do desempenho de cada abordagem usada neste trabalho.

Além das taxas de verdadeiro positivo e falso positivo utilizadas para a construção das curvas ROC, foi utilizada outra medida para avaliar o desempenho dos algoritmos, chamada AUC (*Area Under an ROC Curve*).

4.1. Avaliação do $ACOR_{\mathbb{R}}$ e do AG

Os parâmetros do algoritmo de colônia de formigas estão definidos na tabela 1(a). O valor de N_F, q e k foram definidos a partir dos experimentos realizados em [Socha and Dorigo 2008].

Os testes ocorreram assumindo os seguintes valores para D_0: 15, 20, 25 e 30. A figura 5(a) mostra a curva ROC para cada um desses valores. Nela é possível perceber que o $ACO_{\mathbb{R}}$ tem desempenho semelhante em todos os casos, isso torna-se claro na tabela 1(b), em que destaca-se um aumento máximo de 4% no desempenho do sistema, evidenciado pela AUC. Logo, para todos os valores testados, pode-se considerar que o sistema tem um desempenho aceitável. Porém, levando-se em consideração que aumentar D_0 fará com o sistema seja mais flexível, ou seja, suscetível a erros, o melhor valor para esse parâmetro é 20.

Tabela 1. Parâmetros usados no $ACO_{\mathbb{R}}$ (a) e seu desempenho (b)

(a)

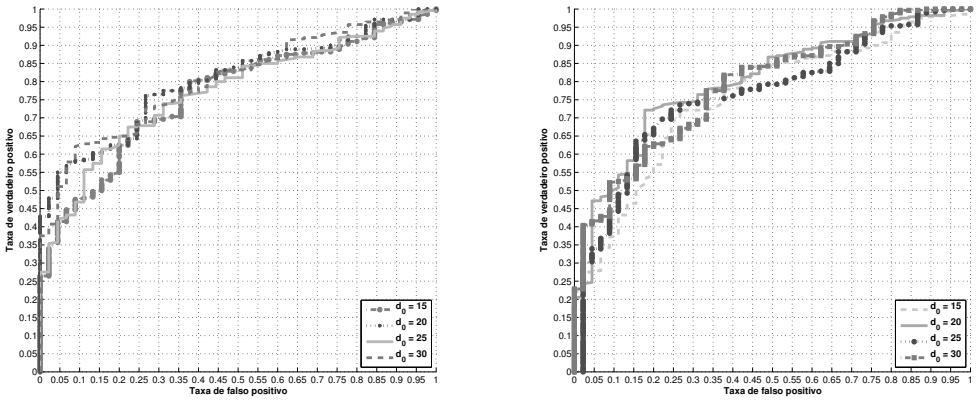
Parâmetros	**Símbolo**	**Valor**
Tamanho do arquivo de solução	k	100
Localidade do processo de busca	q	0,001
Taxa de convergência	ξ	0,85
Número de formigas	N_F	15
Limiar de diferença angular	θ_0	20

(b)

D_0	**Área sob a curva (AUC)**
15	0,765595
20	0,801429
25	0,770913
30	0,802460

Os parâmetros do AG estão apresentados na tabela 2(a), os quais foram baseados em [Pires et al. 2006]. Os valores aplicados a D_0 são os mesmos usados no $ACO_{\mathbb{R}}$. A figura 5(b) ilustra as curvas ROC obtidas pelos resultados do algoritmo genético. Analisando-as, percebe-se que, assim como no $ACO_{\mathbb{R}}$, todas as configurações obtiveram resultados semelhantes. Os resultados de AUC descritos na tabela 2(b) confirmam tal

observação. Dos resultados de AUC obtidos, o melhor resultado foi alcançado quando D_0 = 20, ou seja, AUC de 0,799881. Ao comparar os melhores resultados obtidos pelos dois algoritmos observa-se que o $ACO_{\mathbb{R}}$ tem um desempenho ligeiramente melhor que o AG.



(a) (b)

Figura 5. Resultados obtidos pelo $ACO_{\mathbb{R}}$ (a) e AG (b).

Tabela 2. Parâmetros usados no AG (a) e seu desempenho (b)

(a)

Parâmetros	**Símbolo**	**Valor**
Número de gerações	N_g	100
Tamanho da população	T_p	100
Taxa de cruzamento	Cx	0.8
Taxa de mutação	Mx	0.2
Limiar de diferença angular	θ_0	20

(b)

D_0	**Área sob a curva (AUC)**
15	0,755833
20	0,799881
25	0,763492
30	0,788452

5. Conclusão

Neste artigo foi proposto um estudo comparativo entre duas abordagens bioinspiradas para a resolução do problema da verificação de impressões digitais. Mais precisamente, um Algoritmo Genético e um algoritmo de Colônia de Formigas foram usados para otimizar as variáveis de uma função, que é responsável em realizar o alinhamento das minúcias entre duas impressões digitais, para que as mesmas pudessem ser comparadas.

Os experimentos utilizaram a base de dados FVC 2002, a qual contém imagens com alto grau de distorções, tais como, rotação, translação, escala e ruídos. O processo consistiu na verificação das impressões genuínas, ou seja, imagens de um mesmo dedo são comparadas entre si, e na verificação das impressões impostoras, isto é, comparação entre imagens de dedos diferentes. Os resultados obtidos de cada algoritmo foram expressos por curvas ROC, possibilitando a análise do desempenho de cada abordagem usada neste trabalho.

Observando o critério da AUC as duas abordagens apresentaram resultados satisfatórios, com uma pequena vantagem para o $ACO_{\mathbb{R}}$. A fim de melhorar o desempenho,

pode-se optar por implementar estratégias de otimização complementares às abordagens usadas, como o elitismo, também poderia-se aumentar o tamanho da população de cromossomos ou do arquivo de solução, ou modificar qualquer um de seus atributos. Porém, maior impacto seria causado através da melhoria do processo de extração de minúcias, ou até mesmo o uso de outras características da impressão digital no processo de verificação.

Referências

Associação Brasileira das Empresas de Cartão de Crédito e Serviços (2016). Cartões somam R$ 269 milhões em compras no 1T16, aponta Abecs.

BioLab - University of Bologna (2002). Second international competition for fingerprint verification algorithms.

Dorigo, M., Bonabeau, E., and Theraulaz, G. (1999). *Swarm intelligence: from natural to artificial systems*. Oxford University Press, Inc., New York, NY, USA.

Fawcett, T. (2006). An introduction to roc analysis. *Pattern Recognition Letters*, 27(8):861–874.

Holland, J. H. (1975). *Adaptation in natural and artificial systems*. University of Michigan Press, Ann Arbor.

Jain, A., Hong, L., and Bolle, R. (1997). On-line fingerprint verification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:302–314.

Jain, A. K. and Maltoni, D. (2009). *Handbook of Fingerprint Recognition*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Jain, A. K., Prabhakar, S., and Pankanti, S. (2002). On the similarity of identical twin fingergerprints. *Pattern Recognition*, 35:2653–2663.

Jardini, E. A. (2007). *MFIS: Algoritmo de Reconhecimento e Indexação em Base de Dados de Impressões Digitais em Espaço Métrico*. PhD thesis, Universidade de São Paulo, Escola de Engenharia de São Carlos, Departamento de Engenharia Elétrica.

Michalewicz, Z. (1996). *Genetic Algorithms + Data Structures = Evolution Programs (3rd Ed.)*. Springer-Verlag, London, UK, UK.

Pain, A. K., Ross, A., and Prabhakar, S. (2004). An introduction to biometric recognition. *IEEE Transactions on Circuits and Systems for Video Technology*.

Pires, M. G., Duarte, F. V., and Gonzaga, A. (2006). Verificação de impressões digitais usando algoritmos genéticos. *II Workshop de Visão Computacional (WVC), São Carlos*.

Socha, K. and Dorigo, M. (2008). Ant colony optimization for continuous domains. *European Journal of Operational Research*, 185(3).

Tan, X. and Bhanu, B. (2006). Fingerprint matching by genetic algorithms. *Pattern Recognition*, 39:465–477.

Thai, R. (2003). Fingerprint image enchacement and minutae extraction. *Honours Programme of the School of Computer Science and Software Engineering, The University of Western Australia*.