

Inicialização do Algoritmo do Vagalume Aplicado ao Projeto de Dicionários

Rodrigo Spencer*, Natália Barros†, Paula Amorim‡, Verusca Severo§, Francisco Madeiro¶

*Universidade de Pernambuco-UPE, Recife, Pernambuco

Email: rshc@poli.br

†Universidade de Pernambuco-UPE, Recife, Pernambuco

Email: nataliapaola412@gmail.com

‡Universidade de Pernambuco-UPE, Recife, Pernambuco

Email: paulaavmelo@gmail.com

§Universidade de Pernambuco-UPE, Recife, Pernambuco

Email: verusca.severo@poli.br

¶Universidade de Pernambuco-UPE, Recife, Pernambuco

Email: madeiro@poli.br

Resumo—A quantização vetorial permite reduzir a quantidade de *bits* necessários para representar um sinal. A qualidade do sinal reconstruído depende da qualidade dos dicionários projetados. Em diversos algoritmos, o projeto de dicionário depende da inicialização, ou seja, do dicionário inicial, visto que algoritmos amplamente utilizados para projeto de dicionários começam com soluções iniciais que são melhoradas de forma iterativa. Este trabalho propõe a aplicação de técnicas de inicialização ao algoritmo M-FA-LBG (do inglês, *Modified Firefly Algorithm - Linde-Buzo-Gray*), com o objetivo de alcançar uma melhor qualidade das imagens reconstruídas em comparação à utilização da inicialização aleatória. Os resultados mostram que o uso de técnicas de inicialização aplicadas ao M-FA-LBG permite obter resultados de PSNR (do inglês, *Peak Signal to Noise Ratio*) superiores aos obtidos com inicialização aleatória.

Palavras-chave: Compressão de imagem, quantização vetorial, projeto de dicionários, inicialização de dicionários, algoritmo do vagalume.

I. INTRODUÇÃO

O avanço tecnológico e o crescente uso de serviços onde os dados são produzidos em grande escala levaram ao aumento do volume de dados digitais a serem transmitidos e armazenados. Diante desse cenário, tornam-se cada vez mais necessárias técnicas de compressão de sinais, as quais buscam reduzir o número de *bits* para representar os sinais [1].

Na compressão de sinais, o número de *bits* para representação de um determinado sinal é reduzido, a fim de que os requisitos de memória para armazenamento e de largura de banda para transmissão sejam minimizados [2], [3].

Dentre as técnicas destinadas à compressão de sinais, a QV (quantização vetorial) se destaca por ser amplamente utilizada em compressão de imagem, permitindo o mapeamento de uma sequência de vetores de entrada que pertencem a um espaço de dimensão K em uma sequência de vetores de mesma dimensão, resultando em uma quantidade menor de *bits* para representar o sinal quantizado.

A versão quantizada corresponde a uma versão degradada do sinal original, em que os vetores de entrada são representados pelos correspondentes vetores do dicionário que apresen-

tem maior semelhança [4]. A qualidade do sinal quantizado, portanto, depende da qualidade do dicionário utilizado. Por esse motivo, o projeto de dicionário é um problema relevante no âmbito de QV.

Técnicas de projeto de dicionário têm sido um objeto de estudo amplamente explorado na literatura [5]–[9]. Dentre as técnicas propostas, o algoritmo LBG (Linde-Buzo-Gray) [6] se destaca por ser o mais conhecido, o qual corresponde a um algoritmo de agrupamento que começa com um dicionário inicial que é melhorado de forma iterativa. Com isso, o dicionário inicial é de grande relevância para o desempenho do dicionário final.

Severo *et al.* propuseram em [10] um algoritmo de projeto de dicionário, denominado algoritmo M-FA-LBG (do inglês *Modified Firefly Algorithm - Linde-Buzo-Gray*), obtido a partir de modificações realizadas no algoritmo FA-LBG (do inglês *Firefly Algorithm - Linde-Buzo-Gray*) [5]. Os resultados obtidos com o M-FA-LBG levaram a um desempenho superior na ordem de 1,78 dB sobre o algoritmo LBG para a imagem Clock utilizando dicionários com 512 vetores-código de dimensão 16. O objetivo do presente trabalho é propor uma modificação na etapa inicial do M-FA-LBG, fazendo uso de técnicas de inicialização de dicionário abordadas na literatura [11]–[13], e, assim, alcançar uma melhor qualidade das imagens reconstruídas em comparação à utilização da inicialização aleatória do algoritmo M-FA-LBG.

O restante do trabalho encontra-se organizado como segue. A seção II aborda a QV, contemplando a descrição do algoritmo LBG. As seções III e IV apresentam, respectivamente, os algoritmos do vagalume e M-FA-LBG. A seção V aborda as técnicas de inicialização contempladas neste trabalho. As seções VI e VII apresentam, respectivamente, resultados e conclusões.

II. QUANTIZAÇÃO VETORIAL

Na QV, dado um sinal de entrada X composto por uma sequência de M vetores, $X = \{x_m, m = 1, \dots, M\}$, sendo

estes pertencentes ao espaço euclidiano K -dimensional, \mathbb{R}^K , é realizado um mapeamento de cada vetor x_m em um vetor pertencente a um subconjunto finito W de \mathbb{R}^K constituído por N vetores-código, isto é, $W = \{w_i, i = 1, \dots, N\}$, chamado dicionário [4].

A similaridade entre esses vetores é determinada a partir da utilização de um critério de distância mínima, por exemplo, a distância euclidiana quadrática, definida por

$$d(x_m, w_i) = \sum_{j=1}^K (x_{mj} - w_{ij})^2, \quad (1)$$

em que x_{mj} corresponde à j -ésima componente de x_m e w_{ij} à j -ésima componente de w_i . O vetor x_m será representado pelo vetor código w_i se este apresentar a maior similaridade com x_m dentre todos os demais vetores-código [4].

O projeto de dicionário busca encontrar o grupo de vetores-código representativos para um determinado conjunto de vetores de treino que reduza a distorção introduzida na representação supracitada.

A. Algoritmo LBG

O algoritmo LBG foi proposto por Linde, Buzo e Gray em 1980 [6], sendo também denominado algoritmo de Lloyd generalizado ou GLA (do inglês, *Generalized Lloyd Algorithm*).

Definida a dimensão do quantizador, K , o tamanho do dicionário, N , e o limiar de distorção, ϵ , e sendo n o número de iterações, o algoritmo LBG é assim descrito:

- *Etapa 1* (inicialização): Dado um dicionário inicial W_0 e um conjunto de treino $X = \{x_m, m = 1, 2, \dots, M\}$. Considerar $n = 0$ e $D_{-1} = \infty$;
- *Etapa 2* (particionamento): Seja W_n o dicionário na n -ésima iteração. O espaço \mathbb{R}^K é dividido em N células S_i , denominadas regiões de Voronoi. A partir da regra do vizinho mais próximo, que busca o vetor-código que possua maior similaridade, utilizando uma medida de distância, cada vetor de treino x_m é alocado em uma célula:

$$S_i = \{x : d(x, w_i) < d(x, w_j), \forall j \neq i\}, \quad (2)$$

em que S_i é a i -ésima célula ou região de Voronoi; calcule a distorção

$$D_n = \sum_{i=1}^N \sum_{x_m \in S_i} d(x_m, w_i). \quad (3)$$

- *Etapa 3* (critério de parada): Essa etapa consiste no teste de parada, o qual determina que o algoritmo deve encerrar se $(D_{n-1} - D_n)/D_n \leq \epsilon$, retornando W_n como dicionário final. Caso contrário, deve-se continuar;
- *Etapa 4* (atualização do dicionário): Calcular os novos vetores-código de W_n como os centroides das regiões de Voronoi, como segue

$$w_{ij} = \frac{1}{M_i} \sum_{x_m \in S_i} x_{mj}, \quad (4)$$

em que M_i é a quantidade de vetores de treino alocados na i -ésima região de Voronoi; fazer $W_{n+1} \leftarrow W_n$; Substituir n por $n + 1$ e retornar à *Etapa 2*.

III. ALGORITMO FA

Os algoritmos bioinspirados têm sido vastamente utilizados na busca de soluções de problemas complexos [14], [15]. Dentre esses algoritmos, pode-se citar o algoritmo do vagalume, FA (do inglês, *Firefly Algorithm*), proposto por Yang em 2008 [16].

O algoritmo FA é uma técnica de inteligência de enxame que se baseia na interação entre os vagalumes em seus rituais do acasalamento.

Para a formulação do algoritmo foram consideradas três regras básicas [5], [14], [16]: os vagalumes têm o mesmo sexo, ou seja, todos atraem e são atraídos; a atratividade é proporcional ao brilho emitido, ou seja, quanto maior o brilho emitido, mais atrativo o vagalume será. Além disso, um vagalume de menor brilho vaga em direção a um de maior brilho, bem como aquele que tiver maior brilho vagará aleatoriamente, já que não será atraído por nenhum outro; por fim, o brilho emitido por cada vagalume é proporcional à função *fitness*, ou seja, quanto maior o *fitness*, maior será a emissão de luz.

A intensidade da luz, I , de um vagalume é definida como [14], [17]

$$I(d(v_i, v_j)) = I_0 e^{-\gamma d^2(v_i, v_j)}, \quad (5)$$

em que I_0 corresponde à intensidade da luz quando a distância é nula, γ é o coeficiente de absorção do brilho pelo meio e $d(v_i, v_j)$ é a distância euclidiana entre os vagalumes v_i e v_j .

O fator de atratividade β pode ser formulado como

$$\beta(d(v_i, v_j)) = \beta_0 e^{-\gamma d^2(v_i, v_j)}, \quad (6)$$

em que β_0 corresponde à atratividade do vagalume para $d(v_i, v_j) = 0$ [5].

A posição do vagalume no espaço, a atração deste por um de maior brilho e a aleatoriedade são fatores que regem o movimento dos vagalumes [17]. A Equação (7) determina o movimento que v_i realiza quando segue em direção a v_j

$$v_{iK} = v_{iK} + \beta(d(v_i, v_j))(v_{jK} - v_{iK}) + \alpha \mu_i, \quad (7)$$

em que α corresponde à aleatoriedade do movimento e $\mu_i = (alt - \frac{1}{2})$, com *alt* sendo um número aleatório entre 0 e 1 [5]. Se v_i for o vagalume de maior brilho, então ele se moverá aleatoriamente, como segue

$$v_{iK} = v_{iK} + \alpha \mu_i. \quad (8)$$

O algoritmo FA considera uma população inicial de V vagalumes. Primeiramente, o brilho emitido por cada vagalume é calculado por meio da Equação (5). Posteriormente, calcula-se a atratividade entre os vagalumes por meio da Equação (6). Então, o posicionamento dos vagalumes é atualizado. Cada vagalume se moverá de acordo com a Equação (7). No caso do vagalume de maior brilho, é utilizada a Equação

(8) para determinar sua posição. Caso o critério de parada seja satisfeito, como, por exemplo, o número de iterações, encontra-se a solução que corresponderá ao vagalume de maior brilho. Caso contrário, o processo é repetido.

IV. ALGORITMO M-FA-LBG

Severo *et al.* [10] propuseram em 2016 um algoritmo para projeto de dicionário para QV baseado no FA e no LBG. O algoritmo foi denominado M-FA-LBG (do inglês, *Modified Firefly Algorithm - Linde-Buzo-Gray*), desenvolvido a partir de modificações realizadas no algoritmo FA-LBG proposto por Horng em [5]. Assim como no algoritmo FA-LBG, no M-FA-LBG considera-se que um enxame possui V vagalumes, em que cada um corresponde a um dicionário de tamanho N e dimensão K .

O dicionário projetado, ou seja, o dicionário final, corresponde ao vagalume de maior *fitness*. O *fitness* do vagalume v_a é dado por

$$fitness(v_a) = \frac{1}{D(v_a)}, \quad (9)$$

em que $D(v_a)$ é a distorção para o a -ésimo vagalume.

O algoritmo M-FA-LBG é descrito a partir das seguintes etapas:

- *Etapa 1* (inicialização e definição de parâmetros): O conjunto de treino é dividido em M vetores de dimensão K . A partir deste, são gerados os V dicionários iniciais, todos com o mesmo tamanho N e mesma dimensão K . Em geral, usa-se inicialização aleatória. Além disso, é nessa etapa que α , β_0 , γ e o limiar de distorção são escolhidos. Os parâmetros utilizados neste trabalho foram especificados por Severo *et al.* em [10];
- *Etapa 2* (treinamento via algoritmo LBG e busca pelo vagalume de maior brilho): Para permitir maior influência do conjunto de treino ao projeto de dicionário, o cálculo dos centroides segundo o algoritmo LBG foi introduzido na etapa de atualização do dicionário. Logo, assim como no algoritmo LBG, para cada vagalume, os M vetores do conjunto de treino são alocados nas N regiões de Voronoi e os centroides são calculados. São calculadas as V distorções (uma para cada vagalume) para determinar o vagalume de maior brilho;
- *Etapa 3* (critério de parada e/ou atualização do posicionamento dos vagalumes): A condição de parada é definida pelo limiar de distorção, para o vagalume de maior brilho. Caso não seja atingido, o posicionamento dos vagalumes é atualizado de acordo com as regras estabelecidas para o algoritmo FA (descrito na seção III) e então retorne a *Etapa 2*. A atualização da posição dos vagalumes corresponde à atualização do dicionário.

V. TÉCNICAS DE INICIALIZAÇÃO

O projeto de dicionários para QV tem como objetivo encontrar um dicionário que, utilizado para comprimir o sinal, leve à menor distorção possível.

A inicialização é uma etapa que influencia diretamente o desempenho da QV, porque a velocidade de convergência

e a qualidade do dicionário obtido dependem do dicionário inicial [11].

Uma alternativa de inicialização amplamente utilizada é a aleatória. Há outros métodos de inicialização que podem levar a uma melhor qualidade dos dicionários projetados. Katsavounidis, Kuo e Zhang propuseram em [11] uma técnica de inicialização cuja essência é usar o vetor que é mais diferente dos vetores de treino existentes como vetor-código, pois, segundo os autores, os vetores de treino que estão mais distantes uns dos outros são mais propensos a pertencerem a regiões de Voronoi diferentes. Nyeck e Tossier-Roussey [12] utilizaram o método de máxima entropia para desenvolver uma técnica de inicialização que otimiza o uso dos vetores-código. Hu *et al.* [13] uniram dois métodos de inicialização, descritos em [18] e [19], para gerar um dicionário inicial com melhor qualidade. As técnicas supracitadas serão utilizadas para inicialização do algoritmo M-FA-LBG.

A. Técnica de Inicialização para LBG

A técnica de inicialização de Katsavounidis, Kuo e Zhang [11] se baseia na ideia de que dois vetores de treino com maior distância dentre todos os vetores, apresentam alta probabilidade de pertencerem a regiões de Voronoi diferentes. Levando em conta essa consideração, o método desenvolvido usa o vetor mais diferente dos vetores como o novo vetor-código.

Na primeira etapa do algoritmo, a norma de todos os vetores de treino $X = \{x_m, m = 1, 2, \dots, M\}$ é calculada para que a norma máxima seja encontrada e o vetor x_m que apresentar este valor será definido como o primeiro vetor-código. Então, é calculada a distância entre todos os vetores de treino e o primeiro vetor-código para que o vetor com a maior distância seja definido como o segundo vetor-código. Depois dessa etapa, calcula-se a distância de cada vetor de treino x_m restante ao dicionário, ou seja, $D = d(x_m, w_1) + d(x_m, w_2) + \dots + d(x_m, w_i)$. Após a definição de todas as distâncias, escolhe-se como novo vetor-código o vetor de treino com a maior distância ao dicionário. O processo encerra quando o dicionário tiver tamanho N .

B. Método de Inicialização de Entropia Máxima (MEIM)

As técnicas convencionais de inicialização utilizadas no projeto de dicionário conduzem geralmente a dicionários nos quais alguns vetores-código são subutilizados. No caso extremo, alguns vetores-código podem nunca ser usados. Buscando contornar o problema descrito, Nyeck e Tossier-Roussey [12] desenvolveram o MEIM (do inglês, *Maximum Entropy Initialisation Technique*) com o objetivo de minimizar o erro de quantização. Para tanto, os primeiros vetores-código são escolhidos de tal forma que suas respectivas regiões de Voronoi apresentem tamanhos aproximadamente semelhantes. Com isso, espera-se que o algoritmo convirja rapidamente para um quantizador no qual seus vetores-código contribuam igualmente para a distorção média com um menor erro de quantização [12].

VI. RESULTADOS

No algoritmo, primeiramente, é fornecido um dicionário inicial de tamanho N , $Y_0 = \{y_i, i = 1, \dots, N\}$, com partição $P(Y_0) = \{S_i, i = 1, \dots, N\}$, em que S_i corresponde à i -ésima região de Voronoi do dicionário dado. Seja N_i o número de vetores pertencentes à i -ésima região de Voronoi. Faça $N_i = 1 \forall i$. Seja $X = \{x_m, m = 1, \dots, M\}$ o conjunto de treino. O vetor de treino x_m é alocado na i -ésima região de Voronoi, $x_m \in S_i$, se

$$N_i d(x_m, y_i) < N_j d(x_m, y_j) \forall j \neq i. \quad (10)$$

Se a condição estabelecida na Equação (10) for satisfeita incrementa-se N_i , $N_i = N_i + 1$, e repete-se o processo de alocação para todos os vetores de treino. A etapa seguinte à alocação realiza o cálculo dos centroides de acordo com a Equação (4). Então, o dicionário é definido pelos vetores $Y = \{y_i, i = 1, \dots, N\}$ que mais se assemelham aos centroides.

C. Seleção de Vetores-Código Iniciais Por Meio de Duas Classificações (DSICS)

Com intuito de otimizar dois métodos existentes, um proposto por Yedla, Pathakota e Srinivasa em [18] e o outro por Vimala em [19], Hu *et al.* [13] desenvolveram o método DSICS (do inglês, *Double Sorting-Based Initial Codewords Selection*), dividido em quatro etapas: pré-processamento; ordenamento de forma crescente da distância dos vetores de treino à origem; ordenamento de forma crescente da soma dos valores dos *pixels* dos vetores de treino; e obtenção dos vetores-código iniciais.

Na etapa do *Pré-processamento*, o número de *pixels*, K , em cada vetor de treino é definido, a imagem de entrada é particionada em M vetores de treino compondo o conjunto de treino $X = \{x_m, m = 1, \dots, M\}$, e o tamanho do dicionário, N , é escolhido.

Para a etapa do *ordenamento de forma crescente da distância dos vetores de treino à origem*, faz-se uma cópia de X e a define como C_1 . Então, deve-se encontrar o valor mínimo de *pixel* para cada vetor de treino e subtraí-lo de cada *pixel* do vetor de treino. Em seguida, calcula-se a distância euclidiana de cada vetor treino à origem e ordena-se de forma crescente C_1 de acordo com as distâncias calculadas.

Na terceira etapa, *ordenamento de forma crescente da soma dos valores dos pixels dos vetores de treino*, faz-se uma nova cópia de X e chama-a de C_2 . Então, calcula-se a soma de todos os valores de *pixel* de cada vetor de treino e ordena-se de forma crescente C_2 de acordo com as somas calculadas.

Para a etapa de *obtenção dos vetores-código iniciais*, C_1 e C_2 são divididos igualmente em N conjuntos. Cada par dos conjuntos definidos são chamados c_{1i} e c_{2i} , com $i = 1, \dots, N$. Para cada par de conjunto, se $c_{1i} \cap c_{2i} \neq \emptyset$, o ponto mediano dessa interseção é definido como o i -ésimo vetor-código. Por exemplo, se c_{11} e c_{21} possuem três vetores em comum w_1, w_2 e w_3 , e quando ordenados, se apresentam nessa sequência, $w_1 - w_3 - w_2$, então o vetor w_3 será selecionado como o primeiro vetor-código. Se $c_{1i} \cap c_{2i} = \emptyset$, o i -ésimo vetor-código é determinado como o ponto mediano de c_{1i} .

Utilizaram-se as imagens Barbara, Boat, Clock, Elaine, Goldhill, Lena, Mandrill, Peppers e Tiffany, no formato PGM (do inglês, *Portable Gray Map*), codificadas a 8,0 bpp (256 níveis de cinza), de dimensão 256×256 *pixels* como conjunto de treino no projeto dos dicionários.

Os dicionários iniciais utilizados nas simulações possuem dimensão $K = 16$, isto é, blocos de 4×4 *pixels* e tamanho N igual a 32, 64, 128, 256 e 512 vetores-código, correspondentes às respectivas taxas de codificação iguais a 0,3125 bpp, 0,375 bpp, 0,4375 bpp, 0,5 bpp e 0,5625 bpp.

As versões propostas neste trabalho foram nomeadas de M-FA-LBG (Inic.#1), M-FA-LBG (Inic.#2), M-FA-LBG (Inic.#3) e M-FA-LBG (Inic.#4). Os métodos propostos utilizam o mesmo conjunto de inicializações que foi utilizado pelo M-FA-LBG, no qual os dez dicionários foram inicializados aleatoriamente, sendo que, no M-FA-LBG (Inic.#1) dois dos dez dicionários iniciais foram projetados a partir da técnica de inicialização MEIM e a proposta por Katsavounidis, Kuo e Zhang; no M-FA-LBG (Inic.#2) dois dos dez dicionários iniciais foram projetados utilizando a técnica de inicialização DSICS e a técnica proposta por Katsavounidis, Kuo e Zhang; no M-FA-LBG (Inic.#3) dois dos dez dicionários iniciais foram projetados utilizando as técnicas de inicialização DSICS e MEIM e no M-FA-LBG (Inic.#4) três dos dez dicionários iniciais foram projetados utilizando as técnicas de inicialização DSICS, MEIM e a técnica proposta por Katsavounidis, Kuo e Zhang

Seja u o valor de pico de amplitude da imagem de entrada e H_1 e H_2 duas imagens digitais de tamanho $E_1 \times E_2$ *pixels*, em que H_1 corresponde à imagem original e H_2 à imagem reconstruída (resultado da QV). Para uma imagem codificada a 8,0 bpp (256 níveis de cinza), que apresenta o valor de pico de amplitude $u = 255$, a relação sinal-ruído de pico (PSNR, do inglês, *Peak Signal-to-Noise Ratio*) é definida por

$$\text{PSNR}(\text{dB}) = 10 \log_{10} \left[\frac{u^2}{\text{MSE}} \right], \quad (11)$$

em que o MSE corresponde ao erro médio quadrático entre as imagens H_1 e H_2 , dado por

$$\text{MSE} = \frac{1}{E_1 \times E_2} \sum_{l=0}^{E_1-1} \sum_{c=0}^{E_2-1} [H_1(l, c) - H_2(l, c)]^2, \quad (12)$$

em que $H_1(l, c)$ e $H_2(l, c)$ representam respectivamente os valores de *pixels* das imagens originais e reconstruídas na l -ésima linha e c -ésima coluna.

Para todos os algoritmos, foi utilizado como critério de parada o limiar de distorção $\epsilon = 0,001$. Ao final das simulações, os valores de PSNR médio foram computados. O desempenho em termos de PSNR médio das imagens reconstruídas Barbara, Boat, Clock, Elaine, Goldhill, Lena, Mandrill, Peppers e Tiffany, para cada dicionário de tamanho N projetado usando como conjunto treino a correspondente imagem, é apresentado na Tabela I, onde para cada imagem e para cada tamanho de dicionário, foram destacados em negrito

os melhores resultados. A Tabela II apresenta os resultados obtidos a partir do teste de Wilcoxon.

Os resultados apresentados na Tabela I mostram que o M-FA-LBG (Inic.#1), M-FA-LBG (Inic.#2) e M-FA-LBG (Inic.#4) apresentam resultado superior para todos os tamanhos de dicionário para imagem Clock, para $N \geq 128$ para a imagem Tiffany, para $N = 256$ para a imagem Barbara e para $N = 512$ para as demais imagens, exceto as imagens Barbara, Mandrill e Goldhill, quando comparado ao M-FA-LBG. O M-FA-LBG (Inic.#3) apresenta praticamente o mesmo desempenho da versão original para todas as imagens e tamanhos de dicionário.

Tabela I
PSNR MÉDIO EM DB

Imagem	Método	N				
		32	64	128	256	512
Barbara	M-FA-LBG (Aleat.)	24,83	25,83	26,90	26,88	30,00
	M-FA-LBG (Inic.#1)	24,83	25,83	26,88	28,14	29,98
	M-FA-LBG (Inic.#2)	24,83	25,83	26,90	28,15	29,99
	M-FA-LBG (Inic.#3)	24,83	25,82	26,89	28,20	29,99
	M-FA-LBG (Inic.#4)	24,83	25,83	26,88	28,14	29,98
Boat	M-FA-LBG (Aleat.)	25,02	26,07	27,22	28,61	30,53
	M-FA-LBG (Inic.#1)	25,01	26,08	27,24	28,66	30,73
	M-FA-LBG (Inic.#2)	25,01	26,08	27,24	28,67	30,72
	M-FA-LBG (Inic.#3)	25,01	26,06	27,22	28,61	30,51
	M-FA-LBG (Inic.#4)	25,01	26,09	27,25	28,65	30,74
Clock	M-FA-LBG (Aleat.)	26,68	27,88	29,19	30,71	32,54
	M-FA-LBG (Inic.#1)	26,84	28,25	29,81	32,06	35,68
	M-FA-LBG (Inic.#2)	26,84	28,27	29,81	32,05	35,70
	M-FA-LBG (Inic.#3)	26,64	27,82	29,18	30,71	32,57
	M-FA-LBG (Inic.#4)	26,86	28,28	29,78	32,01	35,66
Elaine	M-FA-LBG (Aleat.)	27,83	29,19	30,54	32,09	33,90
	M-FA-LBG (Inic.#1)	27,84	29,19	30,58	32,15	34,22
	M-FA-LBG (Inic.#2)	27,86	29,17	30,56	32,16	34,23
	M-FA-LBG (Inic.#3)	27,84	29,18	30,54	32,08	33,87
	M-FA-LBG (Inic.#4)	27,85	29,17	30,58	32,15	34,23
Goldhill	M-FA-LBG (Aleat.)	26,76	27,78	28,89	30,19	31,81
	M-FA-LBG (Inic.#1)	26,76	27,78	28,89	30,10	31,84
	M-FA-LBG (Inic.#2)	26,76	27,77	28,90	30,09	31,85
	M-FA-LBG (Inic.#3)	26,77	27,77	28,90	30,18	31,81
	M-FA-LBG (Inic.#4)	26,76	27,77	28,90	30,10	31,81
Lena	M-FA-LBG (Aleat.)	26,68	27,93	29,22	30,79	32,85
	M-FA-LBG (Inic.#1)	26,70	27,91	29,20	30,79	33,16
	M-FA-LBG (Inic.#2)	26,70	27,91	29,20	30,81	33,17
	M-FA-LBG (Inic.#3)	26,69	27,92	29,23	30,81	32,89
	M-FA-LBG (Inic.#4)	26,70	27,93	29,22	30,82	33,18
Mandrill	M-FA-LBG (Aleat.)	23,21	23,85	24,59	25,49	26,73
	M-FA-LBG (Inic.#1)	23,21	23,85	24,59	25,46	26,73
	M-FA-LBG (Inic.#2)	23,20	23,85	24,58	25,47	26,73
	M-FA-LBG (Inic.#3)	23,21	23,85	24,58	25,49	26,74
	M-FA-LBG (Inic.#4)	23,20	23,85	24,58	25,47	26,74
Peppers	M-FA-LBG (Aleat.)	26,20	27,51	28,87	30,54	32,75
	M-FA-LBG (Inic.#1)	26,18	27,48	28,91	30,62	33,07
	M-FA-LBG (Inic.#2)	26,17	27,51	28,88	30,62	33,08
	M-FA-LBG (Inic.#3)	26,18	27,48	28,86	30,54	32,78
	M-FA-LBG (Inic.#4)	26,18	27,50	28,90	30,62	33,05
Tiffany	M-FA-LBG (Aleat.)	29,58	30,61	31,75	33,13	35,01
	M-FA-LBG (Inic.#1)	29,58	30,69	31,93	33,50	35,80
	M-FA-LBG (Inic.#2)	29,58	30,67	31,92	33,50	35,81
	M-FA-LBG (Inic.#3)	29,57	30,60	31,75	33,14	34,99
	M-FA-LBG (Inic.#4)	29,59	30,68	31,92	33,50	35,81

Para a imagem Clock, a diferença de desempenho entre o M-FA-LBG e as versões propostas neste trabalho (Inic.#1, Inic.#2 e Inic.#4) varia de 0,16 dB a 3,16 dB. Para o tamanho de dicionário 32, o M-FA-LBG com a proposta

Inic.#1 e Inic.#2 supera a versão original do M-FA-LBG de inicialização aleatória, em 0,16 dB, já o M-FA-LBG com a proposta Inic.#4 supera o M-FA-LBG em 0,18 dB. Para o tamanho de dicionário 512, o M-FA-LBG com a proposta Inic.#1 supera o M-FA-LBG em 3,14 dB, o M-FA-LBG com a proposta Inic.#2 supera o M-FA-LBG em 3,16 dB, enquanto o M-FA-LBG com a proposta Inic.#4 supera o M-FA-LBG em 3,12 dB. Para a imagem Lena, o M-FA-LBG com as propostas Inic.#1, Inic.#2 e Inic.#4 apresentam praticamente o mesmo desempenho observado com o uso de inicialização aleatória para $N = 32, 64, 128$ e $N = 256$. Para o tamanho $N = 512$, as versões propostas superam o algoritmo original (M-FA-LBG). A diferença de desempenho é de 0,31 dB para Inic.#1, 0,32 dB para Inic.#2 e 0,33 dB para Inic.#4.

Tabela II
TESTE DE WILCOXON PARA O ALGORITMO M-FA-LBG (PSNR).

Imagem	Método	M-FA-LBG				
		N				
		32	64	128	256	512
Barbara	Inic.#1	8,39e-01	8,84e-01	2,46e-01	1,00e+00	1,88e-01
	Inic.#2	8,39e-01	9,03e-01	2,44e-02	1,00e+00	2,44e-02
	Inic.#3	6,52e-01	9,76e-01	5,27e-02	7,22e-01	7,54e-01
	Inic.#4	3,85e-01	8,62e-01	6,15e-01	1,00e+00	7,54e-01
Boat	Inic.#1	4,61e-01	1,88e-01	1,16e-01	6,84e-03	9,77e-04
	Inic.#2	7,84e-01	6,54e-02	1,38e-01	9,77e-03	9,77e-04
	Inic.#3	8,39e-01	7,22e-01	3,85e-01	3,85e-01	8,13e-01
	Inic.#4	7,54e-01	9,67e-02	6,54e-02	4,88e-03	9,77e-04
Clock	Inic.#1	4,88e-03	9,77e-04	9,77e-04	9,77e-04	9,77e-04
	Inic.#2	9,77e-03	9,77e-04	9,77e-04	9,77e-04	9,77e-04
	Inic.#3	7,54e-01	9,58e-01	2,16e-01	4,61e-01	1,16e-01
	Inic.#4	2,93e-03	9,77e-04	9,77e-04	9,77e-04	9,77e-04
Elaine	Inic.#1	9,67e-02	2,78e-01	1,37e-02	6,84e-03	9,77e-04
	Inic.#2	3,22e-02	8,13e-01	1,61e-01	1,95e-03	9,77e-04
	Inic.#3	5,27e-02	6,88e-01	9,68e-01	6,52e-01	9,03e-01
	Inic.#4	3,13e-01	7,54e-01	2,44e-02	1,95e-03	9,77e-04
Goldhill	Inic.#1	6,52e-01	5,00e-01	7,54e-01	1,00e+00	1,37e-02
	Inic.#2	8,84e-01	6,52e-01	3,85e-01	1,00e+00	2,44e-02
	Inic.#3	3,48e-01	7,84e-01	5,00e-01	6,52e-01	9,03e-01
	Inic.#4	4,23e-01	5,39e-01	2,46e-01	1,00e+00	2,44e-02
Lena	Inic.#1	8,01e-02	8,39e-01	9,03e-01	3,48e-01	9,77e-04
	Inic.#2	1,16e-01	7,54e-01	8,13e-01	2,16e-01	9,77e-04
	Inic.#3	1,16e-01	6,15e-01	3,13e-01	1,88e-01	1,16e-01
	Inic.#4	2,16e-01	2,78e-01	5,00e-01	1,16e-01	9,77e-04
Mandrill	Inic.#1	1,86e-02	5,77e-01	7,54e-01	9,98e-01	7,54e-01
	Inic.#2	1,37e-02	2,78e-01	4,23e-01	9,95e-01	3,13e-01
	Inic.#3	4,23e-01	2,78e-01	5,00e-01	4,61e-01	2,16e-01
	Inic.#4	9,03e-01	8,39e-01	6,52e-01	9,86e-01	9,67e-02
Peppers	Inic.#1	9,03e-01	9,90e-01	2,44e-02	9,77e-04	9,77e-04
	Inic.#2	9,03e-01	8,62e-01	1,88e-01	4,88e-03	9,77e-04
	Inic.#3	8,39e-01	8,84e-01	5,00e-01	8,13e-01	9,67e-02
	Inic.#4	6,52e-01	6,15e-01	3,22e-02	2,93e-03	9,77e-04
Tiffany	Inic.#1	3,48e-01	1,37e-02	9,77e-04	9,77e-04	9,77e-04
	Inic.#2	3,48e-01	6,54e-02	9,77e-04	9,77e-04	9,77e-04
	Inic.#3	8,84e-01	7,54e-01	3,13e-01	3,13e-01	7,54e-01
	Inic.#4	1,38e-01	3,22e-02	9,77e-04	9,77e-04	9,77e-04

Para os dicionários de tamanho $N = 32, 64, 128$ e $N = 256$ projetados para as imagens Boat, Elaine e Peppers, todos os métodos propostos apresentam praticamente o mesmo desempenho da versão original. Para o tamanho $N = 512$, os métodos propostos com Inic.#1, Inic.#2 e Inic.#4 superam a versão original (inicialização aleatória). A diferença em relação à inicialização aleatória, para Boat, para $N = 512$, em termos de PSNR de imagens reconstruídas é de 0,20 dB

(M-FA-LBG Inic.#1), 0,19 dB (M-FA-LBG Inic.#2) e 0,21 dB (M-FA-LBG Inic.#4). Tanto para imagem Elaine quanto para a imagem Peppers, o ganho é de 0,32 dB (M-FA-LBG Inic.#1) e 0,33 dB (M-FA-LBG Inic.#2). Para a proposta com Inic.#4, o ganho é de 0,33 dB para a imagem Elaine, e de 0,30 dB para a imagem Peppers.

Para a imagem Barbara, os métodos propostos com Inic.#1, Inic.#2 e Inic.#4 superam a versão original (inicialização aleatória) somente para os dicionários de tamanho $N = 256$, com uma diferença em termos de PSNR de imagens reconstruídas de 1,26 dB com o M-FA-LBG Inic.#1, 1,27 dB com o M-FA-LBG Inic.#2 e 1,26 dB com M-FA-LBG Inic.#4. Para a imagem Tiffany, as versões propostas com Inic.#1, Inic.#2 e Inic.#4 superam o M-FA-LBG para os dicionários de tamanho $N \geq 128$. Para o tamanho de dicionário 256, o M-FA-LBG com a proposta Inic.#1, Inic.#2 e com a proposta Inic.#4 superam a versão original do M-FA-LBG em 0,37 dB. Para o tamanho de dicionário 512, o M-FA-LBG com a proposta Inic.#1, Inic.#2 e Inic.#4 supera o M-FA-LBG em aproximadamente 0,8 dB.

Todos os tamanhos de dicionários projetados com os métodos propostos para as imagens Mandrill e Goldhill apresentam praticamente o mesmo desempenho da versão original.

O teste de Wilcoxon-Mann-Whitney, ou simplesmente Wilcoxon, foi o teste estatístico empregado nas amostras para discriminar qual delas teria o melhor resultado. Definido p - value 0,05, as amostras com valor menor que este são, estatisticamente, diferentes.

Os resultados do teste de Wilcoxon entre o M-FA-LBG e o M-FA-LBG (Inic.#1), M-FA-LBG (Inic.#2) e M-FA-LBG (Inic.#4) apresentados na Tabela II mostram que o valor de p -value apresentou-se sempre menor que 0,05 para a imagem Clock reconstruída para todos os tamanhos de dicionário, bem como para a imagem Tiffany com tamanhos de dicionário $N = 128$, 256 e 512. Para a imagem Lena, o valor de p - value apresentou-se menor que 0,05 para o tamanho $N = 512$. Para a imagem Boat, Elaine e Peppers, o valor de p - value apresentou-se menor que 0,05 para o tamanho $N \geq 256$. Esses resultados sugerem que as amostras que apresentam p - value menor que 0,05 são consideradas diferentes e as versões propostas apresentam desempenho superior em relação ao M-FA-LBG.

VII. CONCLUSÕES

Este trabalho propõe o uso de técnicas de inicialização de dicionário aplicadas ao algoritmo M-FA-LBG visando obter dicionários finais com melhor desempenho em termos da qualidade das imagens reconstruídas.

Com a utilização das técnicas de inicialização, foram obtidos resultados com diferença de PSNR médio em relação ao algoritmo M-FA-LBG com inicialização aleatória na ordem de 3,16 dB para a imagem Clock com $N = 512$. O teste de Wilcoxon entre os resultados de PSNR médio obtidos pelas imagens reconstruídas a partir dos dicionários projetados pelo M-FA-LBG e das versões propostas apresentou valores de p - value menores que 0,05 para diversos casos testados.

Portanto, comprovou-se que é possível obter dicionários de melhor qualidade projetados pelo M-FA-LBG com a utilização de técnicas de inicialização de dicionário.

Como trabalho futuro, pretende-se avaliar a aplicação das técnicas de inicialização estudadas neste trabalho em outros algoritmos bioinspirados aplicados ao projeto de QV, como por exemplo, o algoritmo FSS-LBG (do inglês, *Fish School Search - Linde-Buzo-Gray*) [21].

AGRADECIMENTOS

Os autores agradecem o apoio financeiro do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e da Fundação de Amparo à Ciência e Tecnologia do Estado de Pernambuco (FACEPE).

REFERÊNCIAS

- [1] L. A. Leiva, and E. Vidal, "Warped K-Means: An algorithm to cluster sequentially-distributed data," *Information Sciences*, Elsevier, v. 237, pp. 196–210, July 2013.
- [2] N. S. Jayant e P. Noll, "Digital Coding of Waveforms," *Principles and Applications to Speech and Video*, Englewood Cliffs NJ, USA: Prentice-Hall, 1984.
- [3] P. Noll, "Data Compression Techniques", *1st Working Conference on Common Standards for Quantitative Electrocardiography, Digital ECG Data: Communication, Encoding and Storage*, Leuven, Belgien, pp. 39–57, 1990.
- [4] A. Gersho e R. M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Publishers, Boston, MA, 1992.
- [5] M.-H. Horng, "Vector Quantization Using the Firefly Algorithm for Image Compression," *Expert Systems With Applications*, v. 39, no. 1, pp. 1078–1091, January 2012.
- [6] Y. Linde, A. Buzo, e R. Gray, "An Algorithm for Vector Quantizer Design," *IEEE Transactions on Communications*, v. 28, no. 1, pp. 84–95, January 1980.
- [7] M.-H. Horng e T.-W. Jiang, "Image Vector Quantization Algorithm Via Honey Bee Mating Optimization," *Expert Systems With Applications*, v. 38, no. 3, pp. 1382–1392, March 2011.
- [8] D. Tsolakis, G. E. Tsekouras, A. D. Niros, e A. Rigos, "On The Systematic Development of Fast Fuzzy Vector Quantization for Grayscale Image Compression," *Neural Networks*, v. 36, pp. 83–96, September 2012.
- [9] V. Naumova e K. Schnass, "Fast Dictionary Learning From Incomplete Data," *EURASIP Journal on Advances in Signal Processing*, pp. 1–21, February 2018.
- [10] V. Severo, H. A. S. Leitão, J. B. Lima, W. T. A. Lopes e F. Madeiro, "Modified Firefly Algorithm Applied to Image VQ Codebook Design," *International Journal of Innovative Computing and Applications (IJICA)*, v. 7, p. 202–213, 2016.
- [11] I. Katsavounidis, C.-C. J. Kuo e Z. Zhang, "A New Initialization Technique for Generalized Lloyd Iteration," *IEEE Signal Processing Letters*, v. 1, no. 10, pp. 144–146, 1994.
- [12] A. Nyeck e A. Tossier-Roussey, "Maximum Entropy Initialisation Technique for Image Coding Vector Quantiser Design," *Electronics Letters*, v. 28, no. 3, pp. 273–274, January 1992.
- [13] K.-C. Hu, C.-H. Chen, C.-W. Tsai e M.-C. Chiang, "An Enhanced Initialization Method for Codebook Generation," *IEEE International Conference on Consumer Electronics- Taiwan (ICCE-TW)*, pp. 92–93, 2015.
- [14] X.-S. Yang, *Nature-Inspired Metaheuristic Algorithms*, 2^a ed. University of Cambridge, United Kingdom: Luniver Press, 2010.
- [15] S. Hr. A. Kaboli, J. Selvaraj e N. A. Rahim, "Rain-Fall Optimization Algorithm: A Population Based Algorithm For Solving Constrained Optimization Problems", *Journal of Computational Science*, v. 19, pp. 31–42, 2017.
- [16] X.S. Yang, "Firefly algorithm", *Nature-Inspired Metaheuristic Algorithms*, v. 20, pp. 79–90, 2008.
- [17] I. Fister, I. Fister Jr, X.-S. Yang, e J. Brest, "A Comprehensive Review of Firefly Algorithms," *Swarm and Evolutionary Computation*, v. 13, pp. 34–46, 2013.

- [18] M. Yedla, S. R. Pathakota, T. M. Srinivasa, "Enhancing K-means Clustering Algorithm with Improved Initial Center," *International Journal of Computer Science and Information Technologies*, v. 1, pp. 121–125, 2010.
- [19] S. Vimala, "Techniques for Generating Initial Codebook for Vector Quantization," *3rd IEEE International Conference on Electronics Computer Technology*, v. 4, pp. 201–208, 2011.
- [20] S. Siegel e N. J. Castellan-Jr, *Estatística Não-Paramétrica para Ciências do Comportamento*. São Paulo: Artmed, 2006.
- [21] C. S. Fonseca, F. A. B. S. Ferreira e F. Madeiro. "Vector Quantization Codebook Design Based on Fish School Search Algorithm", *Applied Soft Computing*, v. 73, p. 958–968, 2018.