

Reconhecimento de edifícios utilizando o filtro de gabor, a transformada wavelet e a rede neural perceptron de múltiplas camadas

Buildings recognition using the gabor filter, wavelet transform and multilayer perceptron network

DOI:10.34117/bjdv8n5-164

Recebimento dos originais: 21/03/2022

Aceitação para publicação: 29/04/2022

Leandro Aureliano da Silva

Doutor em Engenharia Elétrica

Instituição: Faculdade de Talentos Humanos – FACTHUS

Endereço: Rua Manoel Gonçalves de Rezende - 230, Uberaba – MG, CEP: 38040-240

E-mail: lasilva@facthus.edu.br

Gilberto Arantes Carrijo

Doutor em Engenharia Elétrica

Instituição: Universidade Federal de Uberlândia – UFU

Endereço: Av. João Naves de Ávila, 2121 - Santa Mônica, Uberlândia

MG, CEP: 38408-100

E-mail: gilberto@ufu.br

Felipe Santos Moreira

Doutor em Engenharia Química

Instituição: Faculdade de Talentos Humanos – FACTHUS

Endereço: Rua Manoel Gonçalves de Rezende - 230, Uberaba – MG, CEP: 38040-240

E-mail: felipe.moreira@facthus.edu.br

Antonio Carlos Lemos Júnior

Mestre em Inovação Tecnológica

Instituição: Faculdade de Talentos Humanos - FACTHUS

Endereço: Rua Manoel Gonçalves de Rezende - 230, Uberaba - MG, CEP:38040-240

E-mail:acjunior@facthus.edu.br

Cleiton Silvano Goulart

Mestre em Física

Instituição: Faculdade de Talentos Humanos – FACTHUS

Endereço: Rua Manoel Gonçalves de Rezende - 230, Uberaba – MG, CEP: 38040-240

E-mail: cleiton.goulart@facthus.edu.br

Lidiana Mendes Sousa

Mestre em Engenharia Elétrica

Instituição: Faculdade de Talentos Humanos – FACTHUS

Endereço: Rua Manoel Gonçalves de Rezende - 230, Uberaba – MG, CEP: 38040-240

E-mail: lidiana.sousa@facthus.edu.br

RESUMO

O reconhecimento de edifícios é uma tarefa bastante difícil, pois as imagens utilizadas possuem ângulos e condições de iluminação diferentes, obstruções parciais de árvores, veículos em movimento ou outros edifícios. Devido a todas essas dificuldades, esse trabalho propõe um sistema de reconhecimento que utiliza a representação de wavelet Gabor para a extração de características, a transformada wavelet para a redução de dimensionalidade e para o reconhecimento a rede neural Perceptron de múltiplas camadas. Para verificar a eficiência desse sistema, utilizou-se um banco de dados de imagens de edifícios de prédios novos e antigos. O desempenho do método proposto foi satisfatório em relação a outros métodos encontrados na literatura.

Palavras-chave: reconhecimento de edifícios, representação de wavelet gabor, extração de características e rede perceptron de múltiplas camadas.

ABSTRACT

The building recognition system proposed in this work shows robust techniques for image recognition under different points of view, lighting conditions or partial obstruction by trees, moving vehicles or even other buildings. This system uses Gabor filtering for feature extraction. Daubechies wavelet transforms to reduce the dimensionality and multilayer perceptron neural networks to perform recognition. In order to assess the behavior of the network, the amount of orientations of the Gabor filter was varied during feature extraction. For testing, we used the building database of Sheffield University. The performance of the proposed method has shown to be satisfactory when compared to other building recognition methods available in the literature.

Keywords: building recognition, gabor filter, feature extraction, multilayer perceptron network.

1 INTRODUÇÃO

Um dos grandes desafios nos dias de hoje em se tratando de reconhecimento de objetos é o reconhecimento de edifícios. O principal objetivo desse reconhecimento é distinguir diferentes edifícios em uma grande escala de imagens. Esse reconhecimento pode ser utilizado em vários tipos de aplicações, como, por exemplo: detecção automática de alvo, projeto arquitetônico, navegação de dispositivos móveis [1] [2], dentre outras.

A grande dificuldade para esse reconhecimento é a enorme variabilidade de parâmetros encontrados na obtenção dessas imagens. Estas variabilidades podem ser: obtenção das imagens em diferentes ângulos, diversas condições de iluminação ou nessas imagens podem ocorrer obstruções parciais de árvores, de veículos em movimento ou de outros edifícios.

Contudo pesquisadores de diversos países vem propondo soluções de sistemas de reconhecimento de edifícios capazes de suplantarem os problemas gerados pelas características fotométricas e geométricas das imagens dos edifícios [1-3].

Conforme afirmam [1] as pesquisas na área de reconhecimento de edifícios são conduzidas em duas abordagens a saber: (i) pesquisas em eficácia que utilizam algoritmos baseados em representação e abordagens baseadas em correspondência de duas imagens; (ii) pesquisas em eficiência que utilizam métodos baseados em redução de dimensionalidade, algoritmos baseados em agrupamento e outros que não utilizam nenhum dos métodos anteriores.

Alguns trabalhos em pesquisas relacionadas à eficácia se destacam, dentre eles, o trabalho de Hutchings e Mayol [2] que desenvolveram um sistema de reconhecimento de edifícios para dispositivos móveis. Esse sistema é utilizado como um guia de turismo. Por exemplo, para uma imagem de consulta, suas características locais são extraídas e descritas respectivamente pelo detector de canto de Harris e pelo descritor *scale invariant feature transform* (SIFT). No processo correspondente (reconhecimento), para cada imagem de consulta conforme a sua posição no sistema de posição global (GPS) é selecionada uma escala. Isso resulta em uma redução do espaço de busca e em um menor custo computacional. No entanto, esse sistema falha quando existem grandes mudanças em relação a diversos pontos de vistas.

Jing Li e Allinson [3] motivados pelas pesquisas em eficiência desenvolveram um sistema de reconhecimento que utiliza extração de características visuais e analisaram como o reconhecimento era influenciado pela aplicação de diferentes algoritmos de redução de dimensionalidade. Eles utilizaram características tais como: cor, forma, textura, intensidade, movimento, dentre outras. Afirmaram, também, que utilizando mais características é possível uma maior precisão para a classificação. No entanto, isso significa uma complexidade computacional maior. Para equilibrar a precisão e o custo computacional, os autores utilizaram apenas três características para o reconhecimento. São elas: a intensidade, a cor e a orientação, respectivamente.

O sistema proposto por eles funciona da seguinte forma: Após a extração dessas características em multi-escala é obtido um conjunto de mapas de características para cada imagem que resulta no modelo de saliência. Depois disso, o modelo de saliência é construído pela divisão de cada característica do mapa para um número de sub-regiões e cada mapa é descrito por uma característica da saliência. Então, para redução de dimensionalidade, essas características de saliências são aplicadas a um desses algoritmos de redução. São eles: análise de componentes principais (PCA), análise de discriminante linear (LDA), análise de discriminante semi-supervisionado (SDA), preservação da projeção local (do inglês: *locality preserving projections*). Essa preservação pode ser um

método supervisionado, como por exemplo, *supervised locality preserving projections* (SLPP) e um método não supervisionado, pode-se citar *unsupervised locality preserving projections* (LPP). Finalmente, aplica-se o resultado a um classificador que no trabalho desses autores, foi o k vizinhos mais próximos.

Em [1] também motivados pelas pesquisas em eficiência, os autores utilizaram um modelo de reconhecimento simples, denominado reconhecimento de edifícios usando filtros orientados. Apesar desses filtros serem utilizados em outros trabalhos foi a primeira vez que eles foram aplicados em tal reconhecimento. As quatro partes do modelo são: (i) a representação das características; (ii) a seleção dessas características; (iii) a redução da dimensionalidade; e (iv) a classificação.

Nesse modelo, os filtros orientados foram utilizados para a representação das características. Na seleção das características foi usada a técnica de pooling e na redução de dimensionalidade a análise de discriminante linear (LDA). Finalmente, para o reconhecimento, eles usaram a máquina de vetor de suporte (SVM).

Seguindo a mesma proposta de Jing Li e Allinson em [1] e [3], esse trabalho tem o interesse de mostrar que apenas a característica de textura é suficiente para o reconhecimento, não sendo necessário utilizar mais de uma característica como mencionaram Jing Li e Allinson em [3]. Para tanto, é utilizado o Filtro de Gabor para a extração dessas características uma vez que esse filtro consegue extrair as características de textura sob diversos ângulos e frequências diferentes, contornando os problemas relacionados às transformações geométricas e fotométricas das imagens utilizadas para o reconhecimento.

Para redução da dimensionalidade do vetor que será gerado pelo Filtro de Gabor, utilizou-se a transformada Wavelet, onde as características, pela análise de multiresolução foram divididas em aproximação e detalhe em múltiplas escalas. Assim, as aproximações permitem identificar as características comuns das imagens de um mesmo edifício. No entanto, os detalhes permitem distinguir um edifício do outro. Nesse sentido, as aproximações dos vetores de Gabor, associado ao mesmo edifício, devem permitir obter um vetor de uma dimensão reduzida, que preserva as informações comuns a todas as imagens desse edifício. Por essa razão é proposto a utilização dos coeficientes de aproximação de wavelet.

Jing Li e Allinson em [3] utilizaram o classificador de k vizinhos mais próximos e em [1] a máquina de vetor de suporte (SVM) como classificadores. Esses classificadores não apresentaram bons resultados em relação à taxa de reconhecimento. Por isso, nesse

trabalho é proposto a utilização da rede *Multilayer Perceptron* (MLP) como classificador. A escolha dessa rede foi em em virtude dela apresentar uma boa capacidade de generalização e ser muito eficiente no reconhecimento de padrões [4]. Cabe salientar que é a primeira vez que ela é utilizada para esse tipo de aplicação.

Diante disso, esse trabalho propõe a integração dessas técnicas esperando que elas se complementem e que permitam uma taxa de reconhecimento superior em relação aos trabalhos de Jing Li e Allinson [1] e [3].

Este artigo está estruturado da seguinte forma: a seção 2 descreve a extração de características usando o filtro de Gabor. A transformada wavelet e a rede *Multilayer Perceptron* são apresentadas nas seções 3 e 4. A seção 5 descreve o método proposto. Os resultados são mostrados na seção 6. E finalmente, na seção 7 são apresentadas as conclusões.

2 EXTRAÇÃO DE CARACTERÍSTICAS USANDO FILTROS DE GABOR

O filtro de Gabor também conhecido como (representação de wavelet Gabor) para duas dimensões é dado por [5-6]:

$$\left. \begin{aligned} \psi_{\Pi}(f, \theta, \gamma, \eta) &= \frac{f^2}{\pi\gamma\eta} e^{-\left(\frac{f^2}{\gamma^2}x_t^2 + \frac{f^2}{\eta^2}y_t^2\right)} e^{j2\pi f x_t} \\ x_t &= x \cdot \cos(\theta) + y \cdot \sin(\theta) \\ y_t &= -x \cdot \sin(\theta) + y \cdot \cos(\theta) \end{aligned} \right\} \quad (1)$$

onde: $j = \sqrt{-1}$, f é a frequência senoidal, θ é a orientação da wavelet, γ é o comprimento da onda senoidal, η é o comprimento de onda do espaço wavelet, que é perpendicular à onda senoidal e (x, y) , é a coordenada dos pixels.

Variando o parâmetro de frequência (f) e a orientação (θ), é obtido um banco de filtros da representação de wavelet Gabor. O parâmetro de frequência (f) e a orientação (θ) são encontrados fazendo-se [5-6]:

$$\left. \begin{aligned} f &= \frac{f_{max}}{(\sqrt{2})^g} \\ \theta &= \frac{h}{8}\pi \\ \psi_{g,h}(x, y) &= \psi_{\Pi}(f, \theta, \gamma, \eta) \end{aligned} \right\} \quad (2)$$

Para obter o banco de filtros da representação de wavelet Gabor os valores e os parâmetros usados em (2) são: $\gamma = \eta = \sqrt{2}$, $f_{\max} = 0,25$, o “g”, responsável pela variação da escala de frequência, variando de 0 a 4 e “h”, responsável pela variação das orientações, variando de 0 a 7 [6].

A imagem filtrada com a representação em duas dimensões da wavelet Gabor, é encontrada convoluindo a imagem original com o banco de filtros. Essa operação é descrita por [6]:

$$O_{g,h}(x,y) = I(x,y) * \psi_{g,h}(x,y) \quad (3)$$

O resultado da convolução obtido utilizando (3), origina sub-imagens com 5 escalas de frequências e oito orientações [6].

O vetor de características (F) é encontrado sub-amostrando com um fator igual a 4 as linhas e as colunas do módulo da imagem filtrada, conforme [6]:

$$F = (O^T_{0,0} \ O^T_{0,1} \ \dots \ O^T_{7,7}) \quad (4)$$

Após obter o vetor de características, ele é normalizado com média zero e variância um. Para uma imagem $m \times n$, F possui dimensão dado por [6]:

$$d = \frac{m \cdot n \cdot g \cdot h}{4 \times 4} \quad (5)$$

onde: m e n são o número de linhas e colunas da imagem original, respectivamente; g é a variação da escala de frequência e h é a variação das orientações.

As funções utilizadas para a representação de wavelet Gabor são senóides complexas bidimensionais modeladas por uma função Gaussiana bidimensional como mostrado em (1). O objetivo dessas funções é extrair as texturas encontradas na imagem descritas pela frequência (f) e orientação (θ).

3 TRANSFORMADA WAVELET

Após a aplicação da extração de características na imagem, é obtido um vetor de características que possui uma dimensão alta. Dessa forma, pensando na redução da dimensionalidade, utilizou-se a transformada wavelet (TW) para reduzir o tamanho desse vetor. A (TW) é definida por [7]:

$$Wf(a, b) = \int_{-\infty}^{\infty} f(t) \psi_{a,b}(t) dt \quad (6)$$

Para um sinal discreto de N pontos, a integral de (6) pode ser aproximada por um somatório, como mostrado em (7):

$$Wf(a, b) = \sum_{t=0}^{N-1} f(t) \psi_{a,b}(t) \quad (7)$$

A função $\psi_{a,b}(t)$, denominada wavelet, é derivada de uma função $\psi(t)$ pela transformação:

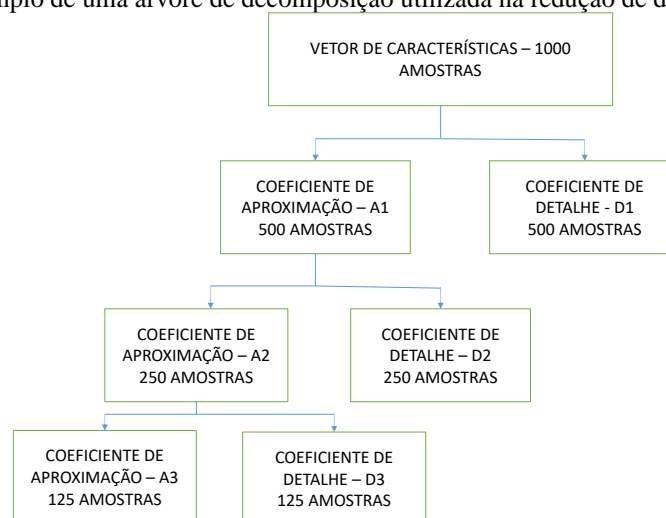
$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \quad (8)$$

onde: “ b ” é a posição ou a translação da wavelet e “ a ” é o parâmetro de escala que está associado à largura da janela.

Existe uma ampla gama de escolhas para a “wavelet mãe” ($\psi(t)$). Dentre elas, pode-se citar: *Daubechies*, *symlets*, *coiflet* e outras [7].

Ao se utilizar a TW para a redução da dimensionalidade do vetor de características, são obtidos coeficientes de aproximação e detalhe. Esses coeficientes originam uma árvore de decomposição. Nessa árvore os coeficientes de aproximação (A_m) representam as informações de baixa frequência associadas a wavelet adotada e os de detalhe (D_m) as informações de alta frequência. A Fig. 1 ilustra a árvore de decomposição utilizada como exemplo de redução de dimensionalidade pela sub-amostragem.

Figura 1. Exemplo de uma árvore de decomposição utilizada na redução de dimensionalidade.



Neste trabalho, a decomposição foi realizada até o nível 3, onde foi utilizado o coeficiente A3 como vetor de características para a rede neural.

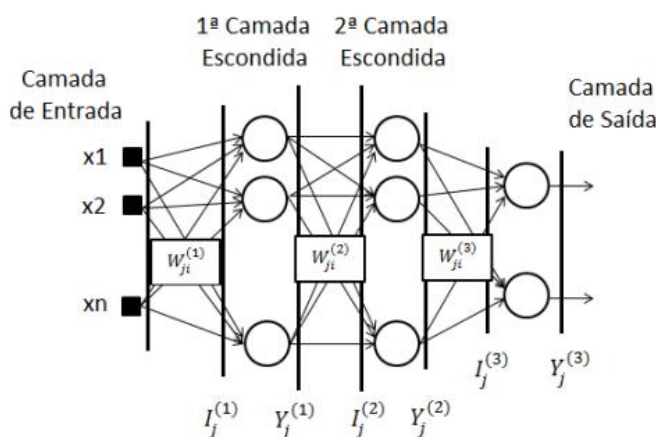
4 REDES MULTILAYER PERCEPTRON

As redes *MLP* consistem em uma camada de entrada, uma ou mais camadas intermediárias e uma camada de saída. Os neurônios que pertencem a camada de entrada são denominados unidades de entrada. Esses neurônios propagam os valores das entradas para as camadas seguintes sem modificação. As camadas intermediárias ou ocultas, transmitem as informações por meio das conexões entre as camadas de entrada e saída. Finalmente, a camada de saída, transmite a resposta da rede neural à entrada aplicada na camada de entrada [8]. A Fig. 2 mostra a arquitetura de uma rede *MLP* com duas camadas intermediárias.

Nessa rede, os neurônios são conectados por arestas e cada aresta está associada a um peso. A comunicação entre os neurônios é realizada por meio dessas interconexões ponderadas. Porém a comunicação ocorre somente entre as unidades de camadas distintas.

Cada neurônio da rede *MLP* possui uma função de ativação. Esta função pode ser linear, logística ou tangente hiperbólica. Este trabalho utilizou a tangente hiperbólica.

Figura 2. Exemplo de uma rede *MLP* com duas camadas escondidas.



Redes *MLP* apresentam variações em relação ao número de camadas, neurônios da camada intermediária, tipo de conexão determinado pelas sinapses e a função de ativação.

A rede *MLP* utiliza um treinamento do tipo supervisionado (‘necessidade de um instrutor’) para indicar a resposta desejada a um padrão de entrada apresentado à ela

durante a fase de aprendizagem. A resposta desejada e aquelas produzidas pelos neurônios de saída são calculadas e subsequentemente utilizadas para ajustar os pesos e os limiares de todos os seus neurônios.

Para o treinamento da rede *MLP*, o método de aprendizagem é o algoritmo *backpropagation*. As fases desse algoritmo são: *forward* e *backward*. Na primeira fase conhecida como *forward*, as entradas são apresentadas e propagadas por meio da rede, camada por camada, calculando a saída de cada neurônio. Durante essa fase, os pesos são fixos e a saída calculada é comparada com a saída desejada, resultando em um erro para cada unidade. Na fase *backward*, o erro calculado é propagado da camada de saída para a camada de entrada e os pesos são ajustados de acordo com regra de correção de erro.

Para uma melhor compreensão do algoritmo *backpropagation* existe a necessidade de definir diversas variáveis auxiliares que serão usadas para o desenvolvimento matemático. A Fig. 2 apresenta um conjunto de variáveis utilizadas para esse entendimento [4].

Considerando a Fig. 2, o desenvolvimento matemático para o potencial de ativação da rede *MLP* é obtido por [4]:

$$I_j^{(1)} = \sum_{i=1}^N W_{ji}^{(1)} \cdot x_i - \theta_j^{(1)} \quad (9)$$

$$I_j^{(2)} = \sum_{i=1}^N W_{ji}^{(2)} \cdot x_i - \theta_j^{(2)} \quad (10)$$

$$I_j^{(3)} = \sum_{i=1}^N W_{ji}^{(3)} \cdot x_i - \theta_j^{(3)} \quad (11)$$

De maneira análoga, o cálculo para as saídas é encontrado por:

$$Y_j^{(1)} = f(I_j^{(1)}) \quad (12)$$

$$Y_j^{(2)} = f(I_j^{(2)}) \quad (13)$$

$$Y_j^{(3)} = f(I_j^{(3)}) \quad (14)$$

onde: $f(.)$ é uma função de ativação.

Após calcular as saídas, o próximo passo é definir o erro entre a saída desejada e a saída da rede. Assim, considerando a k -ésima amostra de treinamento para a topologia ilustrada na Fig. 2, o erro quadrático é definido por [4]:

$$E(k) = \frac{1}{2} \sum_{j=1}^N (d_j(k) - Y_j^{(3)}(k))^2 \quad (15)$$

onde: $Y_j^{(3)}(k)$ é o valor produzido pelo j -ésimo neurônio da saída da rede considerando a k -ésima amostra de treinamento e $d_j(k)$ é o respectivo valor desejado.

A medida do desempenho global da rede para um conjunto de treinamento composto por p amostras pode ser calculada pelo erro quadrático médio:

$$EM = \frac{1}{p} \sum_{k=1}^p E(k) \quad (16)$$

Os pesos da camada de saída podem ser ajustados fazendo-se [4]:

$$W_{ji}^{(3)}(t+1) = W_{ji}^{(3)}(t) + \eta \cdot \delta_j^{(3)} \cdot Y_i^{(2)} \quad (17)$$

onde: η é a taxa de aprendizagem, $\delta_j^{(3)}$ é o gradiente local em relação ao j -ésimo neurônio da camada de saída.

O $\delta_j^{(3)}$ é obtido através de:

$$\delta_j^{(3)} = (d_j - Y_j^{(3)}) \cdot f'(I_j^{(3)}) \quad (18)$$

De maneira análoga, o peso da segunda camada escondida é encontrado fazendo-se [4]:

$$W_{ji}^{(2)}(t+1) = W_{ji}^{(2)}(t) + \eta \cdot \delta_j^{(2)} \cdot Y_i^{(1)} \quad (19)$$

onde: $\delta_j^{(2)}$ é o gradiente local em relação ao j -ésimo neurônio da segunda camada oculta.

O $\delta_j^{(2)}$ é encontrado calculando-se:

$$\delta_j^{(2)} = \left(\sum_{k=1}^N \delta_k^{(3)} \cdot W_{kj}^{(3)} \right) \cdot f'(I_j^{(2)}) \quad (20)$$

É importante ressaltar que os pesos das camadas posteriores dados por (17) são utilizados para atualizar os pesos das camadas anteriores como em (19).

Em relação à primeira camada intermediária, o objetivo do processo de treinamento é ajustar a matriz de pesos $W_{ji}^{(1)}$. Essa atualização é descrita por [4]:

$$W_{ji}^{(1)}(t+1) = W_{ji}^{(1)}(t) + \eta \cdot \delta_j^{(1)} \cdot x_i \quad (21)$$

onde: $\delta_j^{(1)}$ é o gradiente local em relação ao j -ésimo neurônio da primeira camada oculta.

$\delta_j^{(1)}$ é obtido de:

$$\delta_j^{(1)} = \left(\sum_{k=1}^N \delta_k^{(2)} \cdot W_{kj}^{(2)} \right) \cdot f'(I_j^{(1)}) \quad (22)$$

É importante ressaltar que os pesos só são ajustados na fase de treinamento em que são aplicados os passos *forward* e *backward*. Na fase de teste, nenhum tipo de ajuste é efetuado na rede, somente a fase *forward* é processada com o objetivo de gerar as saídas dela.

Diversas variações do algoritmo *backpropagation* têm sido propostas com o objetivo de tornar o processo de convergência da rede mais eficiente. Uma dessas variações é a inserção do termo *momentum* que foi utilizado neste trabalho.

O uso deste termo *momentum*, implica em acelerar a convergência da rede. Formalmente, a atualização da matriz de pesos com esse termo, considerando à L -ésima camada é encontrada fazendo-se [4]:

$$W_{ji}^{(L)}(t+1) = W_{ji}^{(L)}(t) + \alpha(W_{ji}^{(L)}(t) - W_{ji}^{(L)}(t-1)) + \eta \cdot \delta_j^{(L)} \cdot Y_i^{(L-1)} \quad (23)$$

onde: α é a taxa de *momentum* e seu valor varia entre zero e um.

Considerando que o treinamento da rede *MLP* é supervisionado, é necessário que a camada de saída seja composta por vetores binários ou bipolares. Estes vetores formam uma matriz, onde cada coluna corresponde a uma determinada entrada. Os vetores binários e bipolares, são tais que:

$$\vec{V}_{ij} = \begin{cases} 1 & \text{para } i = j \\ 0 & \text{para } i \neq j \end{cases} \quad (2) \quad (4)$$

$$\vec{V}_{ij} = \begin{cases} 1 & \text{para } i = j \\ -1 & \text{para } i \neq j \end{cases} \quad (2) \quad (5)$$

Estudos recentes [9] e [10] mostram que o uso de vetores bipolares ortogonais (OBV)s aumenta a distância euclidiana entre os pontos de saída da rede. Este aumento, beneficia o mapeamento da rede *MLP*, resultando em taxas de acerto maiores e menor tempo de treinamento.

Fausset [11] propôs um algoritmo para gerar os OBVs. Este algoritmo utiliza um vetor de sementes que gera uma sequência de vetores até a determinação final dos OBVs. O número de componentes de cada OBV depende do número de vetores desejados e do número de componentes escolhido para o vetor semente. O número de componentes de cada vetor é dado por:

$$n = 2^k m \quad (26)$$

onde: 2^k é o número de vetores ortogonais entre si para $k > 0$ e m é o número de componentes em um vetor de sementes. É importante ressaltar que o número de vetores é sempre uma potência de 2.

Um conjunto com OBVs é construído com componentes. Os passos do algoritmo de Fausset são:

Passo 1 – Conforme a aplicação, os valores de m e k devem ser determinados pelo usuário. O valor de m inicia igual a 1 e a medida que ele aumenta, a quantidade de OBV's cresce.

Passo 2 – O vetor de sementes é obtido por (27). É importante ressaltar que esse vetor pode ter um único componente.

$$V_m^0 = \overbrace{(1, 1, \dots, 1)}^m \quad (27)$$

Passo 3 – O número de componentes do OBV é dado por (26).

Passo 4 – A partir do vetor de sementes encontrado no passo 3, os vetores $V_{2m}^1 = fcc(V_m^0, V_m^0)$ e $V_{2m}^2 = fcc(V_m^0, -V_m^0)$ são construídos.

onde: $fcc(V_1, V_2)$ é a função de concatenação dos vetores V_1 e V_2 . É importante observar que o vetor de sementes é utilizado duas vezes. Na primeira é realizada a concatenação do V_1 com V_2 . Na segunda vez, o sinal de V_2 é alterado.

Passo 5 – A partir dos vetores V_1 e V_2 obtidos no passo 4, com a utilização da função de concatenação, quatro novos vetores são determinados:

$$V_{4m}^1 = fcc(V_{2m}^1, V_{2m}^1), V_{4m}^2 = fcc(V_{2m}^1, -V_{2m}^1), V_{4m}^3 = fcc(V_{2m}^2, V_{2m}^2) e V_{4m}^4 = fcc(V_{2m}^2, -V_{2m}^2).$$

Os vetores determinados no passo 4 são utilizados duas vezes na função de concatenação. Na primeira o sinal do segundo argumento é mantido e na segunda vez é alterado.

Passo 6 – As concatenações são realizadas até que se obtenha 2^k vetores ortogonais com n componentes $(V_n^1, \dots, V_n^{2^k})$.

5 MÉTODO PROPOSTO

O método proposto consiste em quatro etapas. A primeira realiza o pré-processamento da imagem colorida. Com finalidade de extrair as características da imagem pré-processada, na segunda etapa aplica-se a representação de wavelet Gabor. O

resultado da imagem filtrada usando essa representação é transformado em um vetor de características. Na terceira etapa utiliza-se a transformada wavelet para reduzir a dimensionalidade desse vetor. E finalmente, na quarta etapa são realizados o treinamento e os testes usando a MLP. A Fig. 3 mostra o diagrama em blocos das etapas desse método. A seguir são descritas cada uma dessas etapas.

5.1 PRÉ PROCESSAMENTO

Nessa etapa é realizada a leitura da imagem no banco de dados de edifícios. Após essa leitura, a imagem é convertida para 256 níveis de cinza. A Fig. 4 mostra a imagem original e o resultado obtido nessa etapa.

O objetivo da conversão da imagem para níveis de cinza foi diminuir o tempo de processamento na extração de características.

5.2 EXTRAÇÃO DAS CARACTERÍSTICAS

Na segunda etapa, a extração de características é obtida pela convolução entre a imagem em níveis de cinza e o banco de filtros originado pela representação de wavelet Gabor. Nesta representação, esse banco possui cinco escalas diferentes, e pode-se utilizar até oito orientações e, conforme (1), a coordenada do pixel (x, y) é considerada $(39, 39)$ [12]. A Fig. 5 ilustra o módulo do resultado dessa convolução. Nesta figura é mostrado um exemplo de 40 sub-imagens obtidas usando essas escalas e 8 orientações.

A primeira linha da Fig. 5 ilustra 8 sub-imagens obtidas utilizando a frequência de 0,25 e oito orientações ($0^\circ, 22,5^\circ, 45^\circ, 67,5^\circ, 90^\circ, 112,5^\circ, 135^\circ$ e $157,5^\circ$). Cada uma das outras linhas dessa figura é obtida usando a frequência encontrada em (2) e as mesmas orientações utilizadas nessa primeira linha.

As linhas e as colunas das imagens da Fig. 5 são sub-amostradas por um fator de redução igual a 4. O vetor de características é obtido rearranjando o resultado dessa sub-amostragem em um vetor coluna. Finalmente, para ser utilizado na rede MLP o vetor de características é normalizado com média zero e variância igual a um. Porém, para uma convergência mais rápida da rede, a dimensionalidade do vetor normalizado será reduzida.

A dimensão do vetor de características é obtida usando a Equação (5). Como as imagens utilizadas neste trabalho são 160×120 pixels e são usadas 5 escalas de frequência e 8 orientações neste exemplo, a dimensão desse vetor é igual a 48.000 amostras.

Figura 3. Diagrama em blocos do método proposto.

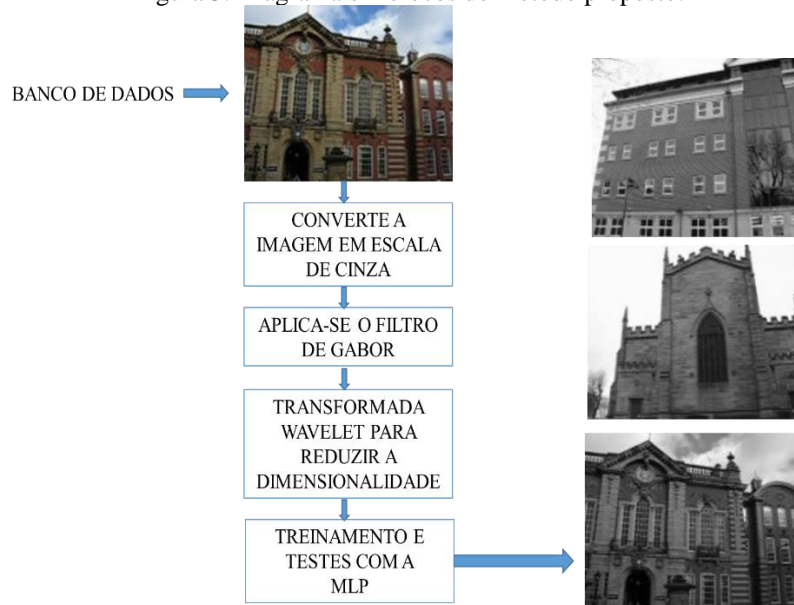


Figura 4. (a) Imagem original; (b) imagem em níveis de cinza

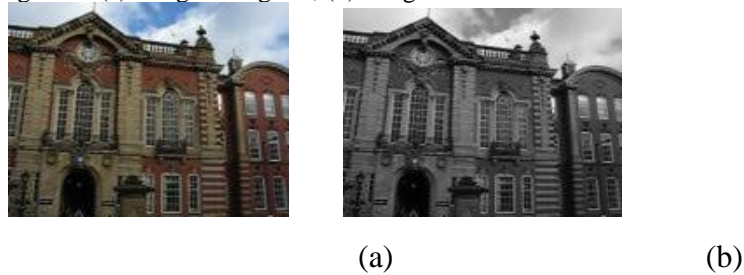
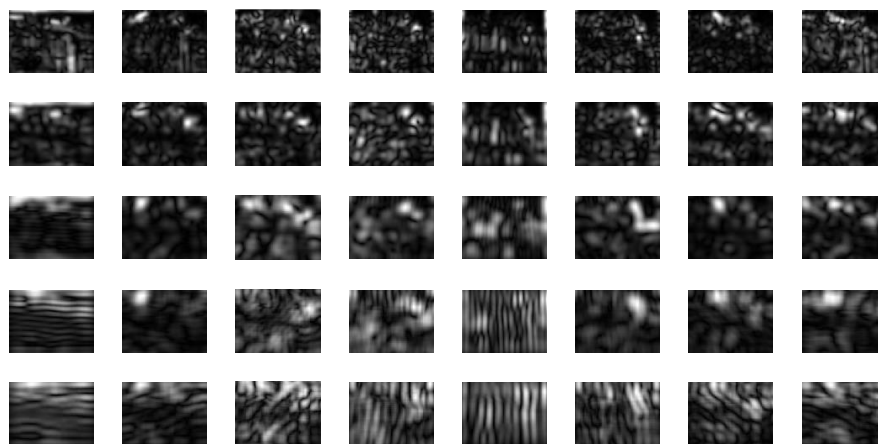


Figura 5. Módulo do resultado obtido da convolução entre a imagem em níveis de cinza e a representação de wavelet Gabor.



5.3 REDUÇÃO DA DIMENSIONALIDADE

Com a finalidade de reduzir a dimensionalidade do vetor de características aplicou-se a transformada wavelet de Daubechies de ordem 5 nesse vetor até o nível 3. O

resultado dessa operação é o novo vetor de características obtido pelo coeficiente de aproximação 3 da TW. É esse vetor que será utilizado para o treinamento e os testes da rede MLP.

5.4 TREINAMENTO E TESTES DA REDE MLP

Para realizar o reconhecimento dos edifícios, foi utilizada uma rede neural artificial MLP. Nesta rede, o número de neurônios da camada de entrada é igual ao tamanho do vetor de características. Considerando a validação cruzada, a única camada intermediária possui 200 neurônios e com a utilização de vetores-alvo do tipo OBV de dimensão 64, a camada de saída possui 64 neurônios.

No treinamento e nos testes com a MLP, foram utilizadas as 80 primeiras imagens de 21 categorias do banco de imagens de edifícios, totalizando 1680 amostras. Deste total, 1344 (64 imagens de cada categoria) foram utilizadas para o treinamento da rede e 336 (16 imagens de cada categoria) para testes. É importante ressaltar que o conjunto de amostras utilizado no treinamento é diferente do conjunto de amostras usado nos testes.

O critério de parada do treinamento da MLP foi o alcance do erro quadrático médio tolerado. Após realizar vários testes, considerou-se as taxas de erro tolerável e aprendizagem iguais a 0,001 e 0,01 [14], respectivamente. Para acelerar a convergência da rede, foi utilizado no treinamento o termo momentum e a taxa de aprendizagem adaptativa.

6 RESULTADOS

Para validar os resultados obtidos apenas com a característica de textura para a classificação dos edifícios, foi utilizado nesse trabalho o mesmo banco de dados usados por Jing Li e Allinson [1] e [3]. Esse banco de dados de imagens é da Universidade de Sheffield SBID [13] e contém 40 categorias de edifícios, totalizando 3.192 imagens no formato JPEG com resolução de 160 x 120 pixels. O SBID combina diferentes variações da imagem em relação a rotação, dimensionamento, diferentes condições de iluminação, oclusões e vibração.

Os algoritmos foram implementados no software MATLAB® R2013b e para a rede MLP foi utilizado o toolbox de redes neurais artificiais desse software. A implementação foi realizada usando uma máquina CoreI5 com 4G de RAM.

Com a finalidade de reduzir o tempo de processamento dos testes realizados foram usadas apenas 21 das 40 categorias do SBID. Para esses testes, foram utilizadas apenas

as primeiras 80 imagens das 21 categorias. Destas imagens, 64 (80%) foram usadas para o treinamento da rede Multilayer Perceptron e 16 (20%) para teste. A configuração dessa rede foi gradiente descendente com o termo momentum igual a 0,8 e taxa de aprendizagem adaptativa, 200 e 64 neurônios nas camadas intermediária e saída, respectivamente e erro desejado de 0,001.

Para verificar o comportamento da MLP foram realizadas simulações variando o número de orientações do Filtro de Gabor e mantendo as 5 escalas de frequência. Após vários testes, para essas simulações adotou-se a transformada wavelet de Daubechies de ordem 5 como a técnica para reduzir a dimensionalidade do vetor de características. A Tabela 1 mostra a taxa de acertos da MLP nessa simulação.

Tabela 1 – taxas de acerto da mlp variando a orientação θ do filtro de gabor usando a tw de daubechies de ordem 5 para reduzir a dimensionalidade.

Nº de orientações θ	Taxa de reconhecimento %	de Nº de Épocas
1	93%	802
2	93%	1043
3	91%	207
4	96%	379
5	93%	160
6	94%	230
7	94%	161
8	95%	227

Pode-se observar na Tabela 1, que a maior taxa de acerto foi de 96% usando 4 orientações na extração de características. Para as outras orientações essas taxas variaram de 91 a 95%.

Em [1] e [3] a maior a taxa de reconhecimento foram respectivamente 94,6% e 85,25%. Cabe salientar que o maior valor obtido pelo método proposto foi de 96%, sendo superior aos valores encontrados por [1] e [3]. A Tabela 2 mostra esse comparativo.

Como já mencionado acima, o método proposto apresentou melhores resultados em relação aos trabalhos de Jing Li e Allinson [1] e [3]. Também é importante lembrar que Jing Li e Allinson combinaram mais de uma técnica para extração de característica, o que exigiu um maior esforço computacional. No entanto, este trabalho adotou apenas a textura para ser utilizada no reconhecimento, o que significou um menor custo computacional e uma maior taxa de reconhecimento.

Tabela 2 – resultado obtido utilizando o método proposto e os resultados encontrados na literatura.

trabalhos correlatos	Base de dados	Taxa de reconhecimento %
Método Proposto	SBID	96%
Jing Li e Allinson [1]	SBID	94,6%
Jing Li e Allinson [3]	SBID	85,25%

7 CONCLUSÃO

Este trabalho apresentou um novo sistema de reconhecimento de edifícios que com apenas uma característica como a textura, conseguiu alcançar uma boa taxa de reconhecimento. Esse novo sistema foi comprado com os trabalhos de Jing Li e Allinson [1] e [3], mostrando resultados satisfatórios. Para fins de comparação, utilizou-se o mesmo banco de imagens que [1] e [3]. Os experimentos foram conduzidos de maneira a verificar como o número de orientações usadas no filtro de Gabor para extração de características, influencia o comportamento da rede MLP.

Nos testes realizados as orientações θ foram variadas em uma escala de 0 a 7. Nestes testes foi possível verificar que a taxa de acertos no reconhecimento usando 4 orientações foi a que apresentou o melhor resultado, ou seja, 96%. Assim, esse resultado quando comparado com os trabalhos de Jing Li e Allinson em [1] e [3] foi o que apresentou a maior taxa de reconhecimento.

REFERÊNCIAS

- [1] J. Li and N. M. Allinson, "Building Recognition Using Local Oriented Features," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 3, pp. 1697-1704, Aug. 2013.
- [2] R. Hutchings and W. Mayol-Cuevas, "Building recognition for mobile devices: Incorporating positional information with visual features," *Comput. Sci., Univ. Bristol, Bristol, U.K., Tech. Rep. CSTR-06-017*, 2005.
- [3] J. Li and N. M. Allinson, "Subspace learning-based dimensionality reduction in building recognition," *Neuro computing*, vol. 73, no. 1-3, pp. 324-330, 2009.
- [4] I. N. Silva; D. H. Spatti; R. A. Flauzino; "Redes Neurais Artificiais para engenharia e ciências aplicadas," *Artliber*, 2010.
- [5] V. Struc and N. Pavesic, "Gabor-Based Kernel Partial-Least-Squares Discrimination Features for Face Recognition," *Informatica*, vol.20(1), pp.115-138, 2009.
- [6] Y. Piparsaniyan, V. K. Sharma, K. K. Mahapatra, "Robust Facial Expression Recognition using Gabor Feature and Bayesian Discriminating Classifier,". *IEEE International Conference on Communicatoin & Signal Processing ICCSP 2014*, 3rd-5th April 2014, Adhiparashakti Engineering College, Melmaruvathur, Tamilnadu.
- [7] Misiti, M; Misiti, Y; Oppenheim, G.; Poggi, J. M; "Wavelet Toolbox User'S Guide," *The Mathworks: Natick*, 1996.
- [8] S. Haykin. "Redes Neurais: Princípios e prática," *Bookman*, 2001.
- [9] J. R. G. Manzan et al. "Improving iris recognition through new target vectors in MLP artificial neural networks," In: *Artificial Neural Networks in Pattern Recognition*. Springer Berlin Heidelberg, 2012. p. 115-126.
- [10] J. R. G. Manzan; S. Nomura, K. Yamanaka. "Mathematical evidence for target vector type influence on MLP learning improvement," In: *International Conference Artificial Intelligence (ICAI)*, Las Vegas, 2012.
- [11] Fausett, Laurene V. "Fundamentals of neural networks". Prentice-Hall, 1994.
- [12] M. Haghighat, S. Zonouz and M. Abdel-Mottaleb, "Identification Using Encrypted Biometrics," *Computer Analysis of Images and Patterns*, Springer Berlin Heidelberg, vol.8048, pp.440-448, 2013.
- [13] 2014 [online]. <http://www.sheffield.ac.uk/eee/research/iel/datasets>
- [14] da Silva Santos, Daiana, and Robson Mariano da Silva. "Aplicação de redes neurais mlp na predição da evapotranspiração de referência." *Brazilian Journal of Development* 6.5 (2020): 25293-25305.