

Aplicação de Redes Neurais Binárias para a Identificação de Anomalias em Produtos Industriais

**Pedro Henrique Rodrigues da Silva, Marco Tulio Gontijo de Sousa,
Pedro Henrique Azevedo de Medeiros, Rodrigo de Oliveira Santos**

¹Curso Ciência de Dados, Pontifícia Universidade Católica de Minas Gerais
Av Dom José Gaspar 500, Belo Horizonte, Minas Gerais, 30535-610

Abstract. *A detecção de anomalias em imagens industriais é um problema crítico para a Indústria 4.0, com aplicações que vão desde a inspeção de qualidade até a manutenção preditiva. No entanto, a implementação de soluções baseadas em deep learning em dispositivos com recursos limitados, como microcontroladores ou GPUs de baixo poder, ainda é um desafio. Este trabalho propõe a aplicação de redes neurais binárias (BNNs) para a detecção de anomalias, visando desenvolver modelos leves e eficientes que possam ser implementados em dispositivos de borda. A relevância do tema é justificada pela necessidade de soluções que combinem alta precisão com baixo custo computacional, além de sua aplicabilidade em cenários industriais reais.*

1. Contextualização

A Indústria 4.0 tem impulsionado a adoção de tecnologias como Internet das Coisas (IoT) e Inteligência Artificial (IA) para otimizar processos e melhorar a qualidade dos produtos [Cheng et al. 2018]. Nesse cenário, a detecção de anomalias em imagens industriais desempenha um papel crucial, permitindo a identificação precoce de defeitos em produtos e reduzindo custos operacionais [Bergmann et al. 2019]. No entanto, a implementação de modelos tradicionais de deep learning em dispositivos com recursos limitados, como microcontroladores ou GPUs de baixo poder, ainda é um desafio significativo [Wang et al. 2020].

Redes neurais binárias (BNNs) surgem como uma solução promissora para esse problema, oferecendo eficiência computacional e leveza ao representar pesos e ativações com valores binários (1 ou -1) [Courbariaux et al. 2015]. Essa abordagem reduz drasticamente o uso de memória e o custo computacional, tornando-a ideal para aplicações em dispositivos de borda [Rastegari et al. 2016]. No entanto, a binarização pode levar à perda de precisão, especialmente em tarefas complexas como a detecção de anomalias [Qin et al. 2020].

Este trabalho propõe a aplicação de redes neurais binárias para a detecção de anomalias em imagens industriais, visando desenvolver modelos leves e eficientes que possam ser implementados em cenários reais. A relevância do tema é justificada pela necessidade de soluções que combinem alta precisão com baixo custo computacional, além

de sua aplicabilidade em cenários industriais reais. A contribuição esperada é avançar o estado da arte em BNNs para aplicações de visão computacional, oferecendo uma solução viável para a indústria moderna.

1.1. Redes Neurais Binárias

Através da necessidade de compressão das redes neurais convolucionais para uso em máquinas convencionais, uma das técnicas que surge como uma solução promissora e rápida é a quantização da rede de pesos e a binarização é a forma mais extrema de realizar essa técnica. Binarização é a técnica onde os dados são convertidos de pontos flutuantes para terem apenas dois valores possíveis: -1(0) ou +1. Isso evita a extensa carga computacional das multiplicações de matrizes de pesos de dados de pontos flutuantes, além de salvar memória, energia e aumentar significativamente a velocidade de treinamento [Qin et al. 2020].

Em uma rede convolucional tradicional, a operação básica é expressa como:

$$z = \sigma(w \otimes a)$$

Onde w e a representam o tensor de pesos e o tensor de ativação gerado pela camada anterior, respectivamente. σ é a função não linear, \otimes representa a função de convolução e z o tensor de saída. A definição popular de uma função de binarização é a seguinte:

$$Q_w(w) = \alpha b_w, \quad Q_a(a) = \beta b_a$$

Onde b_w e b_a são tensores de pesos binários e ativações binárias, com os respectivos escalares α e β . Com as binarizações, o vetor de multiplicação na propagação direta pode ser reformulado da seguinte maneira:

$$z = \sigma(Q_w(w) \otimes Q_a(a)) = \sigma(\alpha\beta(b_w \odot b_a))$$

Onde \odot representa o produto interno para vetores com operações XNOR. No caso da binarização, a propagação retrógrada leva ao desaparecimento do gradiente, devido aos valores serem restritos a 1 e 0. Sendo assim, uma técnica chamada estimador direto, tradução de *straight-trough estimator* (STE) proposta por Hinton *et. al.* resolve o problema do gradiente no treinamento de uma rede binária profunda. A STE é definida como:

$$\text{clip}(x, -1, 1) = \max(-1, \min(1, x))$$

Entretanto, a binarização irá causar um desvio severo de precisão. Mesmo com técnicas apropriadas para aplicação do gradiente descendente, a rede binária falha em retornar uma solução satisfatória. Esse trabalho irá seguir, a princípio, a abordagem utilizada por [McDonnell 2018]

1.2. Embeddings

Embeddings de imagens são representações matemáticas que traduzem características visuais de uma imagem em vetores numéricos de alta dimensão. Esses vetores capturam informações como formas, cores, texturas e até mesmo contextos semânticos, permitindo que algoritmos de aprendizado de máquina processem e analisem imagens de maneira eficiente.

A necessidade de utilizar embeddings de imagens surge especialmente em cenários onde o volume de dados visuais é grande e a análise manual seria inviável. Eles permitem que sistemas automatizados processem imagens de forma escalável e eficiente. A técnica começou a ser amplamente utilizada com o advento de arquiteturas como AlexNet e VGGNet, que demonstraram a eficácia das CNNs em extrair características visuais relevantes. Desde então, modelos mais avançados, como ResNet e EfficientNet, têm refinado a criação de embeddings, tornando-os mais precisos e compactos.

Um embedding de imagem é um vetor de números que representa uma imagem em um espaço multi-dimensional. Cada número no vetor corresponde a um atributo da imagem, por exemplo, cor, forma, textura, etc. O vetor captura a essência da imagem e nos permite comparar com outras imagens. Embeddings também são muito conhecidos como descritores de imagem.

Os embeddings consistem nas principais características da imagem. Quando não existe um número fixo de classes, quando temos muitas classes, quando queremos comparar imagens ou agrupar ou ranquear imagens mais parecidas, os embeddings são muito utilizados.

1.2.1. HOG (*Histogram of Oriented Gradients*)

Desenvolvido por Dalal e Triggs, o Histograma de Gradientes Orientados é um descritor cuja função se dá através da distribuição (histograma) de gradientes orientados de uma imagem. O HOG é capaz de extrair bordas de um ou mais objetos, para depois enviá-las a um algoritmo classificador responsável por definir, por exemplo, se na imagem analisada há ou não uma face [Jr and da Silva 2022].

Gradientes representam a mudança de tonalidade no decorrer da foto em uma orientação. Normalmente, uma mudança abrupta de cor em uma foto representa uma mudança de objeto ou fundo.

Além do HOG, muitos outros descritores se baseiam no uso de gradientes orientados na detecção de objetos; entretanto, este e alguns outros algoritmos utilizam uma técnica chamada contexto de forma (*Shape of Context*, que é a divisão da imagem em blocos para conseguir extrair as bordas de um objeto. Dalal e Triggs se basearam nessa proposta para aplicar um histograma de gradientes orientados em cada célula dessa, distinguindo HOG dos demais descritores [Jr and da Silva 2022].

1.2.2. LBP (*Local Binary Patterns*)

Para explicar o LPB deve-se ter em mente o que é textura em uma imagem. Textura é definida pelas relações locais entre os pixels de uma imagem e capturada por meio de padrões binários. Esses padrões descrevem a variação de intensidade ao redor de cada pixel central e formam a base para a identificação das características texturais da imagem. Desta maneira, o cálculo do operador LBP pode ser visualizado na imagem 1, onde o pixel central comanda a binarização das células ao redor. Aqueles maiores ou iguais ao valor central são convertidos em 1; caso contrário, 0. O código binário LBP deste pixel central é composto dessas marcações de maneira anti-horária. Ao final, o padrão binário é obtido

convertendo o código binário em decimal [Ojala et al. 2002][bo Cai and lu Ying 2009].

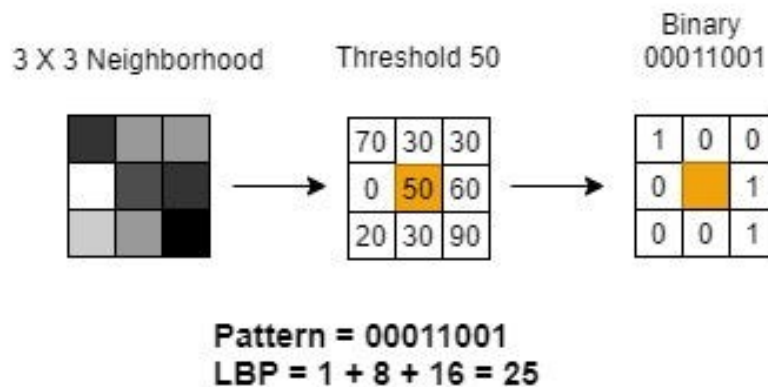


Figure 1. Representação de uma etapas para extração do valor do descritor LBP.

Como pode ser visto na imagem 1, a grade 3x3 é o padrão para a ideia inicial da abordagem. Esse operador não é tão robusto em relação às variações de textura, causadas, por exemplo, por ponto de vista ou direção de iluminação. As variáveis calculadas em um raio 3x3 não capturam estruturas em larga escala. Dessa maneira, para contrapor essa deficiência, é introduzido o operador multi-resolução LBP.

O operador LBP multi resolução [Ojala et al. 2002] recebe como parâmetro para cálculo o raio R de um círculo e os P pixels uniformes contidos neste, como mostra a figura 2. Desta maneira, a influência de P e R devem ser consideradas na performance do operador. Operadores com mais labels produz um histograma longo e uma matriz de distâncias computacionalmente difícil de calcular, por outro lado, um menor número de labels resulta na falta de informação [bo Cai and lu Ying 2009].

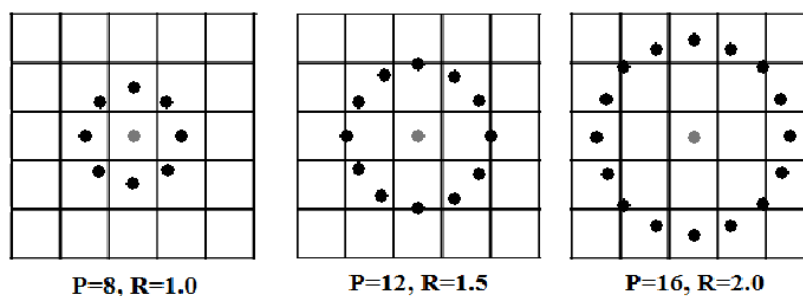


Figure 2. Três operadores LBP circulares simétricos com diferentes raios.

2. Metodologia

References

- Bergmann, P., Fauser, M., Sattlegger, D., and Steger, C. (2019). Mvtec ad—a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9592–9600.
- bo Cai, L. and lu Ying, Z. (2009). Face recognition with locally linear embedding on local binary patterns. In *The 1st International Conference on Information Science and Engineering (ICISE2009)*.
- Cheng, J., Chen, W., Tao, F., and Lin, C.-L. (2018). Industrial iot in 5g environment towards smart manufacturing. *Journal of Industrial Information Integration*, 10:10–19.
- Courbariaux, M., Bengio, Y., and David, J.-P. (2015). Binaryconnect: Training deep neural networks with binary weights during propagations. In *Advances in Neural Information Processing Systems*, pages 3123–3131.
- Jr, D. P. and da Silva, E. J. (2022). Análise de algoritmos de reconhecimento facial: Hog (histogram of oriented gradients) e yolo (you only look once). In *19th International Conference on Information Systems and Technology Management*.
- McDonnell, M. D. (2018). Training wide residual networks for deployment using a single bit for each weight. In *International Conference on Learning Representations (ICLR)*.
- Ojala, T., Pietikainen, M., and Maenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*.
- Qin, H., Gong, R., Liu, X., Bai, X., Song, J., and Sebe, N. (2020). Binary neural networks: A survey. *Pattern Recognition*, 105:107281.
- Rastegari, M., Ordonez, V., Redmon, J., and Farhadi, A. (2016). Xnor-net: Imagenet classification using binary convolutional neural networks. In *European Conference on Computer Vision*, pages 525–542. Springer.
- Wang, Z., Wu, Z., Lu, J., and Zhou, J. (2020). Bidet: An efficient binarized object detector. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2049–2058.