

---

ARTIGO COMPLETO/FULL PAPER

# APLICAÇÃO DE TÉCNICAS DE HASHING E REDES NEURAIS CONVOLUCIONAIS NA BUSCA REVERSA DE IMAGENS

## APPLICATION OF HASHING TECHNIQUES AND CONVOLUTIONAL NEURAL NETWORKS IN REVERSE IMAGE SEARCH

 **Sandy Hoffmann** •  manglesan98@gmail.com

*Instituto Federal Catarinense – Campus Blumenau (IFC)*

 **Ricardo de la Rocha Ladeira** •  ricardo.ladeira@ifc.edu.br

*Instituto Federal Catarinense – Campus Blumenau (IFC)*

**RESUMO.** Neste artigo são abordadas técnicas utilizadas na busca reversa de imagens, abrangendo *hashing* perceptivo e rede neural convolucional, a fim de comparar a taxa de acurácia entre esses métodos. Para os experimentos, foram consideradas imagens manipuladas, possuindo os efeitos de desfoco gaussiano, dessaturação (para escala de cinza), alteração de resolução, realce de nitidez, rotação (90°), espelhamento, corte e inserção de elemento gráfico. O *hashing* perceptivo é um tipo de *hash* que possui menor variação em sua saída quando seu conteúdo de entrada é exposto a pequenas mudanças. No artigo, explora-se sua variação com transformada direta de cosseno. A rede neural convolucional é um tipo de inteligência artificial, geralmente aplicada em imagens, onde as operações de convolução são utilizadas para extraer características específicas dos dados. Esses dados podem ser usados para a busca reversa. Nos experimentos, observa-se que a rede neural convolucional possui mais robustez a alterações como realce de nitidez, corte, rotação e espelhamento. O artigo aborda também uma solução com *hashing* sensível à localidade aleatório, uma técnica de particionamento que utiliza hiperplanos aleatórios para dividir o espaço de busca. É apresentada a aplicação do algoritmo em conjunto com uma rede neural convolucional, a fim de melhorar a assertividade da busca. Este experimento mostrou um aumento na acurácia para as variações das imagens incluindo desfoco gaussiano, dessaturação e recorte. Assim, conclui-se que, em relação ao DCT, a utilização dos vetores característicos extraídos das camadas da CNN é uma alternativa mais robusta, e que o uso do LSH para reduzir o campo de busca mantém a taxa de assertividade.

**ABSTRACT.** This paper explores techniques used in reverse image search, focusing on perceptual hashing and convolutional neural networks (CNNs), aiming to compare their accuracy rates. The experiments involved manipulating images with various effects including Gaussian blur, desaturation (grayscale), resolution reduction, sharpness enhancement, 90° rotation, mirroring, cropping, and the addition of graphic element. Perceptual hashing generates hash values that exhibit minimal changes when the input undergoes slight modifications. The paper examines a variation of perceptual hashing using the discrete cosine transform (DCT). CNN, a type of artificial intelligence typically applied to images, use convolution operations to extract specific features from data. These features can be used for reverse image search. The experiments reveal that CNNs are more resilient to alterations such as sharpness enhancement, cropping, rotation, and mirroring. Additionally, this paper discusses a solution involving random locality-sensitive hashing (LSH), a partitioning technique that uses random hyperplanes to divide the search space. The combination of random LSH and CNNs is presented as a solution to enhance search accuracy. The experiments demonstrating improved accuracy rates for images altered by Gaussian blur, grayscale, and cropping. In conclusion, compared to the DCT, the use of feature vectors extracted from the layers of a CNN constitutes a more robust alternative, and the application of LSH to reduce the search space effectively preserves the accuracy rate.

**PALAVRAS-CHAVE:** Busca Reversa de Imagens • Hashing Perceptivo • Redes Neurais Convolucionais • Hashing Sensível à Localidade

**KEYWORDS:** Reverse Image Search • Perceptual Hashing • Convolutional Neural Network • Locality Sensitive Hashing

---

## 1 Introdução

O grande, crescente e variado volume de dados disponível na internet trouxe também um desafio: localizar informações de maneira rápida e precisa. À medida que o conteúdo digital continua a crescer, encontrar dados relevantes sem mecanismos de busca eficientes se torna impraticável. Nesse contexto, é crescente o

uso de mecanismos de busca através de imagens, tal como oferecem as ferramentas Bing Images<sup>1</sup> e Google Lens<sup>2</sup>. A **RIS (Reverse Image Search – Busca Reversa de Imagens)** é uma técnica que permite identificar e

---

<sup>1</sup> <https://www.bing.com/images>

<sup>2</sup> <https://lens.google.com>

correlacionar imagens com base em suas características visuais, agilizando processos como verificação de originalidade, rastreamento de uso indevido e análise de conteúdo visual em massa [1].

O campo da **RIS** abrange áreas como a busca em mecanismos de pesquisa até a área criminal. Com as técnicas de busca reversa é possível descobrir tentativas de fraude [2], *catfish* em redes sociais [3], falsificação de documentos, violações de direitos autorais [4], entre outras irregularidades.

Um dos principais desafios da busca reversa por imagens é a lacuna semântica, isto é, o processo de transmitir à máquina as características que um ser humano perceberia para interpretar a figura [5]. Além disso, a escalabilidade se torna um problema crítico à medida que o volume de imagens no espaço de pesquisa aumenta, resultando em uma diminuição da eficiência e assertividade do sistema [5]. Outro desafio para os algoritmos de **RIS** é reconhecer imagens manipuladas digitalmente, incluindo alterações como rotação e corte [6], conforme as ilustradas na Figura 1.

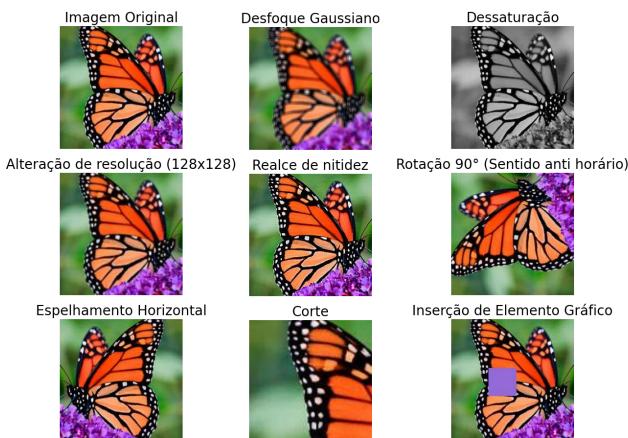


Figura 1. Demonstração de operações de modificação aplicadas a uma imagem original (adaptado de [7]).

Diferentes técnicas podem ser utilizadas para resolver o problema da busca reversa. Este artigo aborda três possibilidades: o algoritmo de *hashing perceptivo*, a **CNN** (*Convolutional Neural Network* – Rede Neural Convolucional), e o uso do **LSH** (*Locality Sensitive Hashing* – *Hashing Sensível à Localidade*) em conjunto com a **CNN**.

O *hashing perceptivo* consiste em gerar representações *hash* de mídias digitais de modo a manter proximidade em seus valores por similaridade [8]. Seu uso se estende a cenários de detecção de conteúdo pirata [9], imagens com teor de abuso infantil [8] e autenticidade de dados [9]. O **LSH** é uma técnica *hash* projetada para

reduzir o custo computacional em espaços de busca com alta dimensionalidade [10].

A **CNN** é um tipo de inteligência artificial amplamente utilizada no processamento de imagens [11], fazendo uso da convolução matemática para aprender os atributos mais marcantes das figuras. A busca reversa é realizada com base nas características extraídas pela rede.

O objetivo do presente artigo é comparar a assertividade entre o *hashing perceptivo* e a **CNN** no campo da busca reversa de imagens, considerando alterações visuais nas figuras. Como forma de melhoria na assertividade dos resultados, é proposto o uso do **LSH** em conjunto com a **CNN**. Seu uso permite particionar os vetores de características da **CNN**, diminuindo o espaço de busca.

## 2 Trabalhos relacionados

Na pesquisa de **Gaillard** [6] é proposto um método para a busca reversa em um conjunto de dados com larga escala utilizando a **CNN** para extrair as características das imagens e o *hash* perceptivo para compactá-las em códigos binários, de modo a manter as similaridades vistas nas imagens. A pesquisa utiliza um conjunto de imagens que sofreram modificações, sendo consideradas alterações de desfoco, dessaturação preto e branco, alteração de resolução, realce de nitidez, rotação e corte.

Os autores **Qazanfari; AlyanNezhadi; Khoshdaregi** [5] se preocupam com a lacuna semântica encontrada no âmbito da **CBIR**, onde critérios como cor e textura podem ter significados diferentes quando interpretados pela máquina em comparação com o usuário. Assim como **Gaillard** [6], os autores **Qazanfari; AlyanNezhadi; Khoshdaregi** [5] ressaltam a capacidade da **CNN** em reproduzir representações mais resilientes contra alterações comuns, especialmente em suas camadas superiores. É demonstrado no estudo de **Gaillard** [6] que a **CNN** tem melhores resultados que o *hash* perceptivo, porém sofre com o desafio da dimensionalidade dos seus vetores de busca, tendo a necessidade de técnicas para melhorar o desempenho quando houver uma grande quantidade de dados.

Assim, os autores **Wekesa; DeCusatis; Zhu** [2] e **Gaillard** [6] citam o algoritmo de **LSH** para permitir o agrupamento de dados. O **LSH** faz o mapeamento das características das figuras em códigos binários, assim facilitando o processo de busca em grandes volumes de dados, mantendo a precisão dos resultados atingida pela **CNN**.

Os trabalhos relacionados discutidos destacam o potencial das redes **CNN** e técnicas de *hashing* para

busca reversa de imagens, além de evidenciar a necessidade de soluções que conciliem precisão e eficiência em cenários com um grande volume de dados.

### 3 Fundamentação Teórica

Esta seção apresenta os principais conceitos necessários para a compreensão dos experimentos propostos nesta pesquisa, focando em três temas principais: *Hashing Perceptivo*, *CNN (Convolutional Neural Network – Rede Neural Convolucional)* e *LSH (Locality Sensitive Hashing – Hashing Sensível à Localidade)*.

#### 3.1 Hashing Perceptivo

O *hashing perceptivo* consiste em gerar uma representação *hash* de modo a manter valores próximos mesmo se a sua entrada for exposta a pequenas mudanças [12]. Este comportamento permite fazer comparações entre os valores gerados a fim de medir a similaridade entre dois *hashes* distintos [6]. Por isso o *hashing perceptivo* entra como estratégia para representar imagens que podem estar expostas a alterações como compressão, conversão de formato, redimensionamento e outras operações.

O *DCT (Discrete Cosine Transformation – Transformada Discreta do Cosseno)* é um algoritmo aplicado no campo do *hashing perceptivo* que utiliza funções do cosseno para somar senoides com diferentes frequências e amplitudes [13]. Esta função possui propriedades de estabilidade a modificações por conta de seus coeficientes de baixa frequência [6], coincidindo com o objetivo base do *hashing perceptivo*.

A fórmula de Hamming é uma alternativa para cálculo de distância no caso do *hashing perceptivo*, podendo ser definida como a Equação 1, onde a função de Hamming *dh* de duas palavras binárias *a* e *b* tendo o comprimento de *n* é definida como o somatório de *i* até *n* utilizando o operador binário *XOR (Exclusive Or – Ou Exclusivo)*. Desta forma, a distância é definida como a soma da quantidade de *uns* resultantes da operação *XOR* entre duas palavras binárias de mesmo comprimento.

$$dh(a, b) = \sum_{i=1}^n (a_i \oplus b_i). \quad (1)$$

#### 3.2 Rede Neural Convolucional

A *CNN (Convolutional Neural Network – Rede Neural Convolucional)* é uma técnica de inteligência artificial que possui a finalidade de processar dados com topologia organizada em grade [11]. É muito utilizada com imagens [14] por conta do uso da convolução no pro-

cesso de aprendizado, reduzindo a quantidade de pesos no processamento.

A aplicação da *CNN* no campo da *RIS* ocorre pela influência do treinamento nas suas camadas, cujos pesos são ajustados a fim de permitir que a rede aprenda a resolver tarefas corretamente, tais como classificações de objetos, por exemplo. As imagens que estão no campo de busca são dadas como entrada na *CNN* e é extraído os vetores de saída de uma das camadas da rede, assim formando uma nova representação para as figuras. Deste modo, a distância entre os vetores pode ser calculada com o objetivo de buscar imagens semelhantes.

O cálculo de similaridade do cosseno é uma forma de estimar a similaridade entre os vetores da *CNN*, medindo o cosseno do ângulo  $\theta$  entre dois vetores *x* e *y*, conforme demonstrado na Fórmula 2 [6]. Desta forma, se os vetores forem similares, o resultado é próximo de 1, pois o ângulo entre os vetores seria igual a 0.

$$sc(x, y) = \cos \theta(x, y) = \frac{x \cdot y}{\|x\|_2 \|y\|_2}. \quad (2)$$

#### 3.3 Hashing Sensível à Localidade

O *LSH (Locality Sensitive Hashing – Hashing Sensível à Localidade)* é um método de particionamento baseado em *hashing*, possuindo a finalidade de encontrar vizinhos próximos em conjuntos de dados com dimensionalidade variada, projetando seus dados em espaços com menores dimensões [15]. O intuito desse algoritmo é reduzir o custo computacional da busca por vizinhos próximos em espaços com uma alta dimensionalidade [10].

A lógica do *LSH* é baseada em partições dos pontos em grupos, assumindo que pontos próximos são mapeados para uma mesma divisão e pontos distantes possuem menor possibilidade de ficarem juntos. O *LSH* baseado em ângulos utiliza projeções aleatórias múltiplas para dividir o espaço de busca, ajustando o ângulo destes de maneira ortogonal.

A Figura 2 ilustra o processo de funcionamento do *LSH* aleatório, mostrando a geração de hiperplanos aleatórios no espaço de dados e uma divisão entre os pontos conforme o ângulo das linhas geradas. Cada ponto é associado a um valor de *hash*, podendo ter colisão com outros vizinhos e assim agrupando dados semelhantes, de maneira a reduzir a dimensionalidade do espaço de busca.

## 4 Materiais e Métodos

Para a realização dos experimentos, foi utilizado um conjunto de dados disponíveis em [7], contendo 100

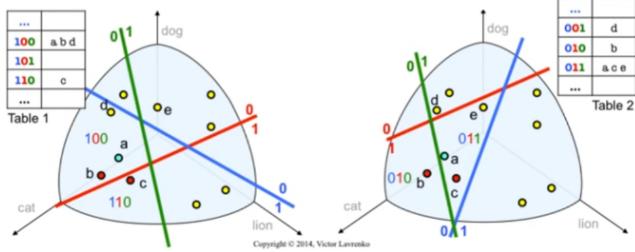


Figura 2. Ilustração do funcionamento do LSH aleatório (adaptado de [16]).

espécies de borboletas e mariposas. Esse conjunto de dados foi escolhido por ter domínio público, e organizado com classes diferentes, o que diversifica o campo de pesquisa de busca reversa. As imagens são padronizadas no formato JPEG, com o tamanho de 224x224x3. São 500 imagens avaliadas, de modo que para cada figura foram criadas oito novas, cada uma com uma das alterações propostas, sendo estas:

- Desfoco Gaussiano (raio=2)
- Dessaturação (Preto e Branco)
- Alteração de resolução (128x128)
- Realce de nitidez
- Rotação (90° em sentido anti-horário)
- Espelhamento (horizontal)
- Corte - canto inferior direito (112x112x3)
- Inserção de elemento gráfico (quadrado) em ponto aleatório da figura

A escolha das operações de desfoco gaussiano, dessaturação preto e branco, realce de nitidez, rotação e corte é dada por elas representarem casos básicos de alterações visuais geralmente encontradas na internet [6]. O efeito de espelhamento horizontal também é uma técnica muito utilizada, principalmente para aumentar dados no processo de treinamento de redes [17]. A inserção de elemento gráfico é escolhida por ser comum em anúncios falsos ou conteúdos ofensivos [18]. Após a geração das imagens alteradas, o espaço de busca foi composto por 4500 imagens. A forma de avaliação dos experimentos propostos segue a seguinte lógica: para cada imagem consultada, consideram-se as nove imagens mais similares retornadas pelo algoritmo, e espera-se que como resultado se tenha entre essas nove, a própria imagem somada das suas versões modificadas. Cada vez que a consulta retornar uma dessas variações, é contada como acerto, e caso contrário, será considerada um erro. Nos experimentos, foram separadas etapas de avaliação para cada operação visual, considerando uma quantidade fixa de imagens consultadas.

Nos experimentos propostos, é avaliada a acurácia dos algoritmos em relação às imagens de consulta, con-

siderando as modificações visuais propostas. O objetivo é medir o impacto dessas alterações sobre a eficácia da técnica apresentada, permitindo uma análise detalhada de como elas influenciam os resultados obtidos. Todos os experimentos implementados utilizaram como base a linguagem de programação Python, na versão 3.12.1, e estão disponíveis *online* no repositório do Github<sup>3</sup>.

O primeiro experimento compara o método **DCT** com a **CNN** em termos de acurácia, considerando a quantidade de acertos obtida por cada técnica. O **DCT** implementado utilizou do método fornecido pela biblioteca OpenCV<sup>4</sup>. Os vetores de características extraídos da **CNN** utilizaram do modelo pré treinado VGG16, pertencente ao *framework* Keras<sup>5</sup>.

No segundo experimento, acrescentado à busca reversa pela **CNN**, é proposto o uso do **LSH** para partitionar o espaço de busca dos seus vetores. Seu uso é recomendado para atingir melhorias de acurácia e eficiência da busca [19]. Para fazer os testes foi implementado o **LSH** aleatório, e indexados em sua estrutura os vetores resultantes da **CNN**, sendo parametrizado com 100 tabelas e *hashes* com comprimento de 8 bits.

## 5 Resultados

Esta seção discute os resultados obtidos ao longo da pesquisa, nos dois experimentos apresentados na Seção 4.

### 5.1 Experimento Comparativo entre DCT e CNN

Para este experimento, foram selecionadas 50 imagens de cada categoria de alteração, incluindo também imagens sem modificações. A Tabela 1 retrata a porcentagem de acertos do **DCT** e da **CNN**.

Nota-se, conforme a Tabela 1, que o **DCT** obteve melhores resultados nos casos de desfoco gaussiano e de imagens com dessaturação (preto e branco), conseguindo um aumento percentual de, respectivamente, 26,3% e 57,8 %, se comparado à **CNN**. A **CNN** conseguiu resultados superiores nos casos de buscas por imagens com método original, mudança de tamanho e com o elemento gráfico adicionado, com um aumento percentual médio de 12,4%. Nos casos de realce de nitidez, corte, rotação e espelhamento, a **CNN** conseguiu ganhos expressivos, obtendo um aumento de, 21,3%, 55,8%, 382,0% e 573,9%, respectivamente, se comparados aos resultados obtidos pelo **DCT**. Assim, é obtido resultados

<sup>3</sup> [https://github.com/SandyHoffmann/BuscaReversaImagens\\_CNN-DCT-LSH](https://github.com/SandyHoffmann/BuscaReversaImagens_CNN-DCT-LSH)

<sup>4</sup> [https://docs.opencv.org/4.x/d2/de8/group\\_\\_core\\_\\_array.html#ga85aad4d668c01fdbd64825f589e3696d4](https://docs.opencv.org/4.x/d2/de8/group__core__array.html#ga85aad4d668c01fdbd64825f589e3696d4)

<sup>5</sup> <https://keras.io/api/applications/vgg/>

Tabela 1. Comparação de Acurácia entre DCT e CNN.

Método	DCT (%)	CNN (%)
Original	66,6%	73,1%
Desfoque Gaussiano	65,7%	52,0%
Dessaturação (Preto e Branco)	66,6%	42,2%
Alteração de Resolução	66,4%	76,2%
Realce de Nitidez	66,6%	80,8%
Rotação (90%)	11,7%	56,4%
Espelhamento	11,5%	77,5%
Corte	11,1%	17,3%
Inserção de Elemento Gráfico	65,7%	74,2%

semelhantes ao trabalho de [Gaillard](#) [6], confirmando que a [CNN](#) é robusta contra rotação e corte, e segue a tese de [Qazanfari; AlyanNezhadi; Khoshdaregi](#) [5], que diz que a [CNN](#) tem mais resiliência contra operações comuns. É somado a essas observações o alto desempenho da mesma com o espelhamento, operação que o [DCT](#) não mostrou robustez.

## 5.2 Aplicação do LSH em conjunto com a CNN

Este experimento sorteia 100 figuras de cada tipo de alteração visual. As porcentagens da Tabela 2 são referentes à comparação do uso somente da [CNN](#) e da aplicação do [LSH](#) nos vetores de características da [CNN](#), possuindo como entrada o mesmo conjunto de imagens, e demonstrando sua porcentagem de acerto.

Tabela 2. Comparação de Acurácia entre CNN e LSH.

Método	CNN (%)	LSH (%)
Original	73,3%	80,4%
Desfoque Gaussiano	49,2%	77,7%
Dessaturação (Preto e Branco)	44,8%	71,0%
Alteração de Resolução	77,6%	81,4%
Realce de Nitidez	77,0%	75,9%
Rotação (90%)	55,0%	61,4%
Espelhamento	78,3%	74,8%
Corte	17,2%	34,2%
Inserção de Elemento Gráfico	76,4%	81,4%

O [LSH](#) mostrou-se efetivo para melhorar a taxa de acurácia da [CNN](#), superando os números de acertos em sete das nove categorias possíveis. Nas categorias realce de nitidez e espelhamento, os resultados ficaram próximos, com a [CNN](#) conseguindo mais acertos por uma margem de 1,4% e 4,6%, respectivamente. O uso do [LSH](#) teve destaque nas transformações de desfoque gaussiano, dessaturação (preto e branco) e corte, conseguindo respectivamente um aumento percentual de

57,9%, 58,4% e 98,8% se comparados aos resultados obtidos apenas pela [CNN](#). Assim, comparando diretamente o uso da [CNN](#) com a aplicação do [LSH](#), é suportada a teoria de [Gaillard](#) [6], mostrando que a aplicação da técnica [LSH](#) mantém a acurácia dos resultados.

## 6 Conclusão

Com o presente trabalho, foi possível observar técnicas para efetuar a busca reversa por imagens, destacando o uso da [CNN](#) e do *hashing perceptivo DCT*. Foi medida a acurácia das técnicas perante modificações nas imagens, simulando cenários que podem ocorrer dentre um mecanismo de busca. Por fim, foi proposto o uso do [LSH](#) em conjunto com a [CNN](#), sendo esta uma técnica usada para particionamento do espaço de busca e que auxilia em questões de assertividade na busca dentro do conjuntos de dados.

É ressaltada a importância da busca reversa de imagens, que atua como ferramenta em diversos campos, desde mecanismos de pesquisa até soluções voltadas à segurança computacional. Entre suas aplicações estão a detecção de fraudes, a violação de direitos autorais, a falsificação de documentos, entre outras.

Com os resultados obtidos foi possível observar que se comparada ao *hashing perceptivo DCT*, a [CNN](#) teve resultados mais robustos, o que pode ser explicado pelo processo de aumento de dados geralmente utilizado na fase de treinamento, onde são feitas transformações visuais nas imagens [6]. O uso do [LSH](#) aleatório para limitar o espaço de busca da [CNN](#) manteve a precisão obtida e melhorou a taxa de acertos.

Para pesquisas futuras, destaca-se a ampliação dos experimentos conduzidos no trabalho, utilizando um conjunto de imagens maior e mais diversificado. Mais um ponto a ser explorado é a diversificação das técnicas utilizadas, como a implementação de outros algoritmos de *hashing perceptivo* e [LSH](#), bem como o uso de outros *frameworks* de [CNN](#) treinadas.

A aplicação dos algoritmos de busca reversa em detecção de *deepfakes* é mais um fator que deseja-se explorar futuramente com essa pesquisa. A ideia central acerca desse novo experimento é detectar se um vídeo foi manipulado a partir de conteúdo original ou se é autêntico.

Outro critério a ser abordado em pesquisas futuras é o desempenho, visto que a busca reversa de imagens geralmente atua sob um grande conjunto de dados, o que torna o tempo um fator fundamental.

## Referências

- 1 Wan, G. G.; Liu, Z. Content-based information retrieval and digital libraries. *Information Technology and Libraries*, 2008. ISSN 0730-9295, 2163-5226. DOI: [10.6017/ital.v27i1.3262](https://doi.org/10.6017/ital.v27i1.3262). Disponível em: <https://hdl.handle.net/1969.1/183318>.
- 2 Wekesa, E. N.; DeCusatis, C.; Zhu, A. A Black Box Comparison of Machine Learning Reverse Image Search for Cybersecurity OSINT Applications. *Electronics*, v. 12, n. 23, 2023. ISSN 2079-9292. DOI: [10.3390/electronics12234822](https://doi.org/10.3390/electronics12234822). Disponível em: <https://www.mdpi.com/2079-9292/12/23/4822>.
- 3 Asian and Pacific Training Centre for ICT for Development (APCICT). *Course on Trust and Security Using Digital Technologies*. Institutional Repository - ESCAP, 2024. Web. Disponível em: <https://hdl.handle.net/20.500.12870/7079>.
- 4 George, A. S.; Baskar, D.; Pandey, D. Establishing Global AI Accountability: Training Data Transparency, Copyright, and Misinformation. *Partners Universal Innovative Research Publication (PUIRP)*, v. 02, n. 03, jun. 2024. DOI: [10.5281/zenodo.11659602](https://doi.org/10.5281/zenodo.11659602). Disponível em: <https://doi.org/10.5281/zenodo.11659602>.
- 5 Qazanfari, H.; AlyanNezhadi, M. M.; Khoshdaregi, Z. N. *Advancements in Content-Based Image Retrieval: A Comprehensive Survey of Relevance Feedback Techniques*. 2023. arXiv: [2312.10089 \[cs.CV\]](https://arxiv.org/abs/2312.10089). Disponível em: <https://arxiv.org/abs/2312.10089>.
- 6 Gaillard, M. *Perceptual Hashing using Convolutional Neural Networks for Large Scale Reverse Image Search*. INSA, 2017. Disponível em: <https://github.com/mgaillard/MasterThesis/blob/master/Thesis.pdf>.
- 7 Piosenka, G. *Butterfly and Moths Image Classification 100 species*. 2022. Accessed: 2024-10-04. Disponível em: <https://www.kaggle.com/datasets/gpiosenka/butterfly-images40-species/data>.
- 8 Farid, H. *An Overview of Perceptual Hashing*. Journal of Online Trust e Safety, 2021. Disponível em: <https://tsjournal.org/index.php/jots/article/view/24/14>.
- 9 Khelifi, F.; Bouridane, A. Perceptual Video Hashing for Content Identification and Authentication. *IEEE Transactions on Circuits and Systems for Video Technology*, v. 29, n. 1, p. 50–67, 2019. DOI: [10.1109/TCSVT.2017.2776159](https://doi.org/10.1109/TCSVT.2017.2776159).
- 10 Rajaram, S.; Scholz, M. Client-friendly classification over random hyperplane hashes. In: SPRINGER. JOINT European Conference on Machine Learning and Knowledge Discovery in Databases. 2008. P. 250–265.
- 11 Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*. MIT Press, 2016. <http://www.deeplearningbook.org>.
- 12 Du, L.; Ho, A. T.; Cong, R. *Perceptual hashing for image authentication: A survey*. ELSEVIER, 2020. Disponível em: <https://doi.org/10.1016/j.image.2019.115713>.
- 13 Zauner, C. *Implementation and Benchmarking of Perceptual Image Hash Functions*. v. 43. 2010. Disponível em: <https://api.semanticscholar.org/CorpusID:17075066>.
- 14 Weidman, S. *Deep Learning from Scratch: Building with Python from First Principles*. O'Reilly Media, Incorporated, 2019. ISBN 9781492041412. Disponível em: <https://books.google.com.br/books?id=PRSCwwEACAAJ>.
- 15 Jafari, O. et al. A Survey on Locality Sensitive Hashing Algorithms and their Applications. *CoRR*, abs/2102.08942, 2021. arXiv: [2102.08942](https://arxiv.org/abs/2102.08942). Disponível em: <https://arxiv.org/abs/2102.08942>.
- 16 Lavrenko, V. *LSH.9 Locality-sensitive hashing: how it works*. 2015. YouTube. Accessed: 2024-10-07. Disponível em: <https://youtu.be/Arni-zkqMBA?si=u-zEdudQaWhuJhh5>.
- 17 Alomar, K.; Aysel, H. I.; Cai, X. Data Augmentation in Classification and Segmentation: A Survey and New Strategies. *Journal of Imaging*, v. 9, n. 2, 2023. ISSN 2313-433X. DOI: [10.3390/jimaging9020046](https://doi.org/10.3390/jimaging9020046). Disponível em: <https://www.mdpi.com/2313-433X/9/2/46>.
- 18 Samanta, P.; Jain, S. Analysis of Perceptual Hashing Algorithms in Image Manipulation Detection. *Procedia Computer Science*, v. 185, p. 203–212, 2021. Big Data, IoT, and AI for a Smarter Future. ISSN 1877-0509. DOI: <https://doi.org/10.1016/j.procs.2021.05.021>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1877050921011030>.
- 19 Schiavo, A. et al. *Sketches image analysis: Web image search engine using LSH index and DNN InceptionV3*. 2021. arXiv: [2105.01147 \[cs.CV\]](https://arxiv.org/abs/2105.01147). Disponível em: <https://arxiv.org/abs/2105.01147>.