Aplicação de métodos de meta-aprendizado para recuperação de dados complexos

Application of meta-learning methods for recovery of complex data

Thalia Akemi Kojo
thaliakj@gmail.com
Universidade Tecnológica Federal do Paraná, Cornélio Procópio, Paraná, Brasil

Pedro Henrique Bugatti
pbugatti@utfpr.edu.br
Universidade Tecnológica Federal do Paraná, Cornélio Procópio, Paraná, Brasil

RESUMO

Este projeto busca analisar uma nova abordagem para melhorar o processo de recuperação de imagens baseada conteúdo, utilizando a metodologia de meta-aprendizado para que os algoritmos de classificação supervisionada (e.g., *Random Forest, Naive Bayes*) possam aprender a identificar o descritor de imagens mais adequado para maximizar a eficácia na extração de características relevantes das imagens. Para corroborar a presente proposta utilizou-se o conjunto de imagens públicas *Corel-1000*. Para o referido conjunto a abordagem proposta detectou automaticamente que os descritores com os melhores resultados foram o *Color and Edge Directivity Descriptor (CEDD)*, o *Fuzzy Color and Texture Histogram (FCTH)*, e o *Border/Interior Pixel Classification (BIC)*. Para tanto, foram utilizadas como pré-características para o meta-aprendizado histogramas de cor e histogramas monocromáticos, utilizados na composição dos vetores de características, sendo que os mesmos produziram resultados semelhantes. Com relação aos classificadores, *Random Forest* teve uma performance superior ao *Naive Bayes*, mas evidenciou uma forte tendência de favorecer a classe mais dominante.

PALAVRAS-CHAVE: Processamento de imagens. Aprendizado do computador. Inteligência artificial.

ABSTRACT

This project aims to test a new approach to improve the content-based image retrieval process through a meta-learning methodology. To do so, we used supervised classifiers (e.g., Random Forest and Naive Bayes) to learn how to detect the most suitable image descriptor to maximize the efficacy of feature extraction. To corroborate our approach, we performed experiments using a public image dataset (Corel-1000). Regarding this dataset, our approach detected that the descriptors with the best results were the Color and Edge Directivity Descriptor (CEDD), the Fuzzy Color and Texture Histogram (FCTH), and the Border-Interior Pixel Classification (BIC). As meta-features we used the color histograms and monochromatic histograms, and they presented similar results. With respect the classifiers, the Random Forest had a superior performance than the Naive Bayes. However, it showed a strong tendency to favor the most dominant class.

KEYWORDS: Image processing. Machine learning. Artificial intelligence.

Recebido: 19 ago. 2020.
Aprovado: 01 out. 2020.

INTRODUÇÃO

O processo de recuperação de imagens baseada em conteúdo, do inglês *Content-based image retrieval (CBIR)* se tornou relevante, com o crescente interesse na procura de informações por meio de similaridades, pois o método de busca de dados utilizando palavras-chaves, etiquetas ou rótulos se tornou não ideal. Ainda mais em relação a recuperação de imagens, que de acordo com Pareek (2017, p.1509-1513), retornam muitos dados irrelevantes, como também palavras chaves adicionadas por humanos, tais métodos podem não descrever uma imagem adequadamente.

Khoker (2012) diz que diferentes imagens podem possuir características mais relevantes para a identificação e classificação de seu conteúdo, e o uso correto destas características aumenta a performance de um sistema CBIR, juntamente com a escolha ideal do algoritmo de cálculo de similaridade . Ou seja, um meio de aumentar a performance deste sistema seria identificar, para cada imagem, as características essenciais e relevantes, que obteriam o melhor desempenho utilizando uma determinada métrica de distância.

Mas realizar esta identificação para todas as imagens a serem pesquisadas, por meio da aplicação de vários extratores de características diferentes, é um processo demorado e inviável. Porém, torna-se plausível mediante a aplicação do método de metalearning. Lemke (2013) afirma a possibilidade de transmitir o conhecimento de um sistema de aprendizagem como experiência para outro, sendo assim possível identificar o melhor extrator de características por meio do treino de um algoritmo de aprendizagem, a partir de um treino já realizado, reduzindo assim o tempo.

MATERIAIS E MÉTODOS

Para testar nossa proposta foi utilizado o conjunto de imagens público *Corel-1000*, composto por 1000 amostras, contendo 10 classes com 100 imagens cada. A implementação de um sistema CBIR, e a análise de sua performance, foram realizadas na linguagem de programação *Python*. Já o treino dos algoritmos de *metalearning*, e sua análise de performance, foram realizados no software WEKA.

Como já mencionado na introdução, CBIR é um método de recuperação de imagens baseada em conteúdo, na qual uma imagem a ser pesquisada tem suas características extraídas e comparadas com as características extraídas de outras imagens pertencentes a uma *database*. Sendo que, a imagem ou o conjunto de imagens, que possuem maiores graus de similaridades com a imagem pesquisada, são retornadas, de acordo com Suresh (2008).

Essas características são extraídas por meio do uso de descritores de imagem, onde Long (2003) afirma que esses algoritmos de extração de informações visuais são responsáveis por definir a essência de uma imagem, extraídos a partir de pontos chaves em uma imagem, sendo estas características baseadas na cor, textura, forma ou relação espacial.

Para que o sistema saiba o quão semelhante uma imagem é em relação às outras, são utilizadas medidas que indicam seu grau de similaridade, calculado a partir de funções de distâncias entre vetores, como a distância euclidiana, que

conforme a pesquisa de Khoker (2012) além dela ser a métrica mais utilizada, ela é eficaz e possui baixa complexidade computacional.

Dado D a distância euclidiana entre o vetor de características da imagem I a ser pesquisada, e o vetor de características da imagem DT presente em uma *database*, onde n equivale ao tamanho destes vetores, o cálculo desta distância é realizado conforme a Eq. (1).

$$D(I, D_T) = \sqrt{\sum_{k=1}^{n}(I(k) - D_T(k))^2} \quad (1)$$

Cada imagem tem suas características extraídas pelos descritores indicados no quadro 1, e armazenadas em uma database. Após, é aplicado o algoritmo de *K-Nearest Neighbors (KNN)*, em conjunto com métrica da distância euclidiana. Com base em Asmita (2017) este algoritmo tem por premissa que dados que possuem seus valores ou vetores próximos, são similares.

Quadro 1 – Descritores utilizados para extração de características

Descritores utilizados	
Auto Color Correlogram	*Joint Composite Descriptor (JCD)*
Border/Interior Pixel Classification (BIC)	*Local Binary Patterns (LBP)*
Color and Edge Directivity Descriptor (CEDD)	*Local Color Histogram (LCH)*
Fuzzy Color and Texture Histogram (FCTH)	*Moments*
Gabor	*Multiparameter Optimization (MPO)*
Global Color Histogram (GCH)	*Modified Phase-Only Correlation (MPOC)*
Haralick	*Pyramid Histogram of Oriented Gradients (PHOG)*
Haralick Color	*Reference Color Similarity*
Haralick Full	*Tamura*

Fonte: Autoria própria (2020).

Os descritores que obtiveram as melhores performances por meio do cálculo da precisão, revocação e *f-measure*, foram selecionados para o treino de um modelo de *metalearning*, na qual este modelo tentou classificar para cada imagem o descritor que obteve a melhor performance.

Neste projeto, foram utilizados os algoritmos *Random Forest* e *Naive Bayes*, que conforme Asmita (2017) é um algoritmo que contém várias árvores de decisões, na qual cada árvore treina individualmente, mas o resultado leva em consideração a predição de cada uma delas, e um algoritmo de classificação probabilístico baseado na probabilidade condicional, utilizando o Teorema de *Bayes*, mas necessita que as características assumidas sejam independentes entre si, respectivamente. Esses foram utilizados para geração do modelo de aprendizado, o qual visou definir o descritor ideal, ou seja, o que obteve a maior performance, para os vetores de características utilizados.

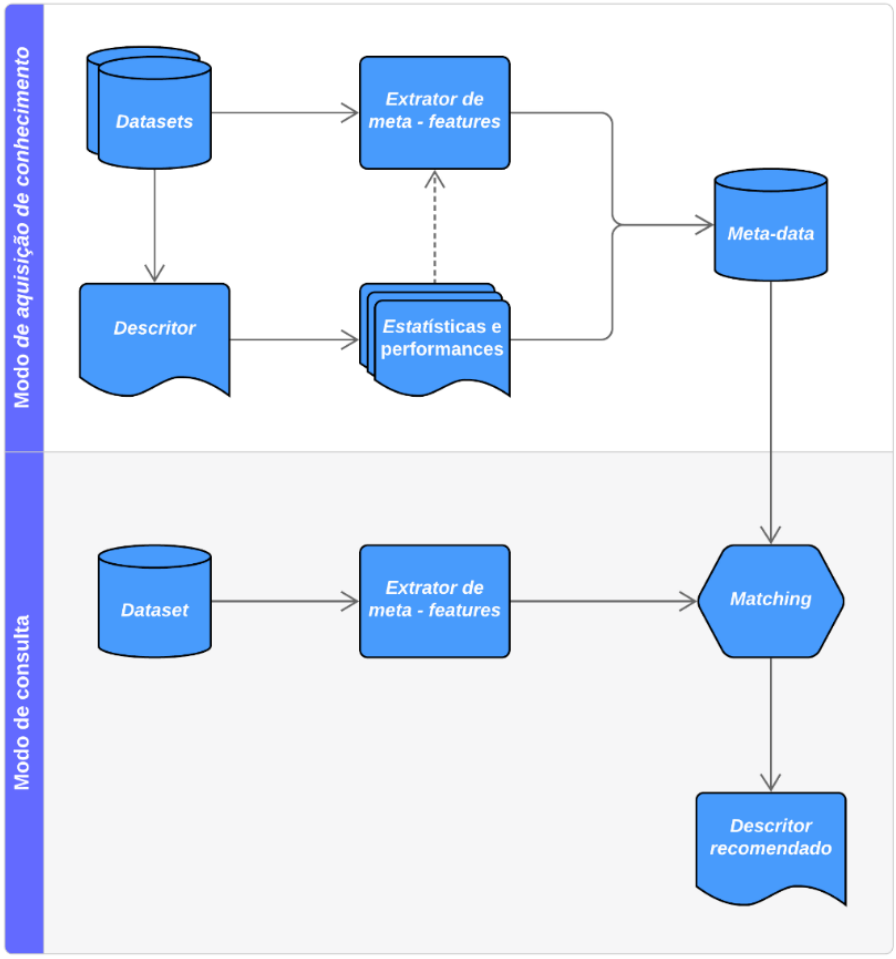
Vetores de características são vetores compostos a partir de pontos chaves de uma imagem, com o rótulo da classe na qual esta imagem pertence, que nesta proposta foram utilizados dois vetores diferentes compostos por:

a) Histograma da imagem monocromática (*grayscale*) + nome do descritor ideal.

b) Histograma da imagem colorida (RGB) + nome do descritor ideal.

A Figura 1 representa uma arquitetura simples do funcionamento da classificação do descritor ideal adaptada do modelo proposto por SOUZA (2008). O processo é divido em duas partes: aquisição de conhecimento, e consulta do descritor recomendado.

Figura 1 – Arquitetura de um metalearning de classificação



Fonte: Autoria própria.

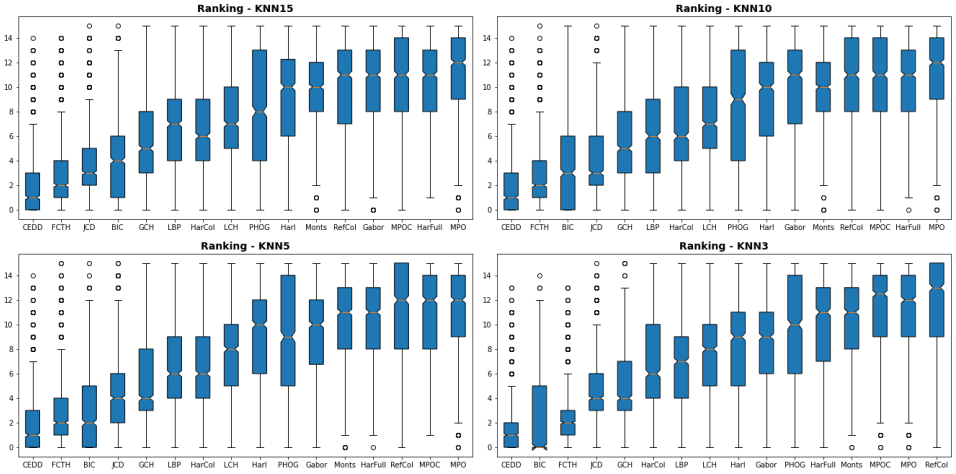
A fase de aquisição, consiste na aquisição de dados que relacionam a performance dos descritores aplicados em relação aos *datasets* de entrada. Estes *datasets* tem suas características extraídas e armazenadas em vetores, como também tem suas informações extraídas pelos descritores aplicados a eles, na qual sua performance é calculada por meio de medidas estatísticas. O vetor composto pela junção dessas características extraídas com o rótulo do descritor ideal, formam os vetores de características, também chamados de *meta-features*, e o conjunto desses vetores geram o *meta-data*.

Na fase de consulta do melhor descritor, um novo *dataset*, em que se deseja identificar o descritor recomendado, tem suas características extraídas pelo mesmo extrator da fase anterior. *Meta-features* do novo *dataset* passam por um processo de *'Matching'* com a *meta-data*, ou seja, um processo onde um *meta-learner* é utilizado para gerar uma função de aproximação mapeando estas *meta-features* com os descritores utilizados. A partir desta função, a abordagem é capaz de identificar qual descritor teria a melhor performance para este *dataset*.

RESULTADOS E DISCUSSÕES

Utilizando o conjunto de imagens *Corel-1000*, foi realizado o *ranking* da performance de cada descritor utilizando as métricas de precisão, revocação e *f-measure*. Os melhores descritores, baseados na precisão, uma vez que os *rankings* baseados na revocação e *f-measures* produziram resultados semelhantes, podem ser observados na Figura 3, para as consultas aos 15-NN, 10-NN, 5-KNN e 3-NN, respectivamente.

Figura 3 – *Boxplot* do ranking da performance dos descritores



Fonte: Autoria própria.

Inicialmente, foram escolhidos os quatro melhores descritores para a composição dos vetores de características, realizando o treino de um modelo com os classificadores *Random Forest* e *Naive Bayes*, com um classificador randômico como *baseline* para verificar o desempenho do modelo, na *database* utilizando o KNN-15 e usando ambos vetores de características, ou seja, o monocromático e o colorido. Tais resultados podem ser vistos no Quadro 2, juntamente com as matrizes de confusão do melhor resultado do *Random Forest* e do *Naive Bayes* presentes nos Quadros 3 e 4, respectivamente.

Quadro 2 – Quadro de precisão dos classificadores utilizando quatro descritores

KNN-15	Randômico [%]	*Random Forest* [%]	*Naive Bayes* [%]
Monocromático	24,9	58,3	58,3
Colorido	25,0	61,8	32,9

Fonte: Autoria própria (2020).

Quadro 3 – Matriz de confusão do R. Forest

	BIC	FCTH	JCD	CEDD
BIC	36	5	0	96
FCTH	7	19	1	152
JCD	6	1	2	92
CEDD	11	9	2	561

Fonte: Autoria própria (2020).

Quadro 4 – Matriz de confusão do N. Bayes

	BIC	FCTH	JCD	CEDD
BIC	0	0	0	137
FCTH	0	0	0	179
JCD	0	0	0	101
CEDD	0	0	0	583

Fonte: Autoria própria (2020).

Analisando os resultados do classificador *Naive Bayes*, percebe-se que ele obteve uma precisão maior que a randômica em ambos vetores, e uma precisão maior para o monocromático em relação ao colorido. Porém, sua matriz de

confusão mostra que o modelo seguiu a classe majoritária, ou seja, ele prediz a classe com a maior probabilidade de ocorrência.

Já os resultados do classificador *Random Forest*, percebe-se que ele também obteve uma precisão maior que a randômica em ambos vetores. No entanto, ao contrário do anterior ele produziu uma precisão maior para o colorido e sua matriz de confusão indica que, mesmo levemente seguindo a regra majoritária, este conseguiu alcançar uma precisão maior.

Após isto, foi feito o mesmo treino seguindo os padrões anteriores, mas ao invés de utilizar os quatros melhores descritores, foram utilizados apenas os dois melhores, para o *dataset* classificado com as consultas 15-KNN, 10-KNN, 5-KNN e 3-NN, e seus resultados podem ser observados no Quadro 5.

Quadro 5 – Quadro de precisão dos classificadores utilizando dois descritores

KNN	Meta-feature	Randômico [%]	*Random Forest* [%]	*Naive Bayes* [%]
15	Monocromático	49,9	74,5	49,9
	Colorido	50,0	74,3	49,9
10	Monocromático	50,0	74,6	47,7
	Colorido	50,1	74,9	47,7
5	Monocromático	49,9	79,7	79,9
	Colorido	49,9	79,3	46,9
3	Monocromático	50,0	86,4	86,4
	Colorido	49,9	86,5	47,9

Fonte: Autoria própria (2020).

A performance do *Naive Bayes* foi igual, senão inferior à *baseline* randômica, e inferior ao *Random Forest* em todos os testes, excluindo os dois dados discrepantes (79,9% e 86,4%), que resultaram nestes valores usando a classe majoritária. Em contrapartida, o *Random Forest* obteve em todos os testes uma performance acima do randômico e melhor do que o do *Naive Bayes*, mesmo aparentando possuir um resultado superior à medida que o KNN diminui, é importante levar em consideração e observar as matrizes de confusão. Por exemplo, as do vetor colorido para cada um dos KNNs, segundo os Quadros 6 a 9, respectivamente dos 3-KNN, 5-KNN, 10-KNN e 15-KNN.

Quadro 6 – Matriz de confusão do KNN-3

	CEDD	BIC
CEDD	863	1
BIC	134	2

Fonte: Autoria própria (2020).

Quadro 7 – Matriz de confusão do KNN-5

	CEDD	BIC
CEDD	790	7
BIC	200	3

Fonte: Autoria própria (2020).

Quadro 8 – Matriz de confusão do KNN-10

	CEDD	BIC
CEDD	727	19
BIC	232	22

Fonte: Autoria própria (2020).

Quadro 9 – Matriz de confusão do KNN-15

	CEDD	BIC
CEDD	727	18
BIC	239	16

Fonte: Autoria própria (2020).

Por meio desta investigação enxerga-se que mesmo que os valores de K menores tenham produzido uma precisão maior, estes valores se dão pelo maior número de dados pertencentes a classe dominante, e que à medida que o KNN

aumenta, tendência é uma maior dispersão na predição de classes neste classificador.

Em relação aos vetores de características, estes não mostraram claramente um ser superior ao outro, visto que os melhores resultados não variaram seguindo um padrão conforme o aumento do K, e nem mostraram uma diferença significativa entre eles.

CONCLUSÃO

Nesta proposta foi desenvolvido uma abordagem baseada em *metalearning* para a definição do melhor descritor de imagens de forma automática. Para tanto, foram testados o histograma de imagens monocromáticas e o histograma de imagens coloridas RGB para a composição dos *meta* vetores de características. Os descritores, *auto color correlogram, border/interior pixel classification (BIC), color and edge directivity descriptor (CEDD), fuzzy color and texture histogram (FCTH), gabor, global color histogram (GCH), haralick, haralick color, haralick full, joint composite descriptor (JCD), local binary patterns (LBP), local color histogram (LCH), moments, multiparameter optimization (MPO), modified phase-only correlation (MPOC), pyramid histogram of oriented gradients (PHOG), reference color similarity* e *Tamura* foram utilizados para a extração das características das imagens, e por fim, o *Random Forest* e *Naive Bayes* como o classificadores.

Analisando todas as informações obtidas foi deduzido, por meio dos experimentos, para o conjunto de imagens *Corel-1000*, que os descritores com melhores resultados foram o CEDD, o FCTH, e o BIC. Ambos *meta* vetores compostos pelos histogramas produziram resultados semelhantes. O classificador *Naive Bayes* forneceu resultados iguais ou piores que a *baseline* randômica, salvo quando seguiu a classe majoritária. Já o *Random Forest* produziu os melhores resultados, mas evidenciou uma forte tendência de favorecer a classe mais dominante.

AGRADECIMENTOS

Agradeço à Universidade Tecnológica Federal do Paraná por proporcionar o equipamento e ambiente propício para o desenvolvimento da minha iniciação científica.

REFERÊNCIAS

ASMITA SINGH, MALKA N. HALGAMUGE AND RAJASEKARAN LAKSHMIGAN THAN. Impact of Different Data Types on Classifier Performance of Random Forest, Naïve Bayes, and K-Nearest Neighbors Algorithms. In: **International Journal of Advanced Computer Science and Applications(ijacsa)**, 8(12), 2017.

HALL EIBE FRANK, GEOFFREY HOLMES BERNHARD PFAHRINGER PETER REUTEMANN MARK; H. WITTEN, IAN. The weka data mining software: An update. sigkdd explorations.v. 11, 2009

KHOKER, AMANDEEP; TALWAR, RAJNEESH. Content-based image retrieval: Feature extraction techniques and applications. In:. [S.I.: s.n], 2012

LEMKE, Christiane; BUDKA, Marcin; GABRYS, Bogdan. Metalearning: a survey of trends and technologies. **Artificial Intelligence Review**, DOI: 10.1007/s10462-013-9406-y, 06 2013.

PAREEK, S.; MANDORIA, H. L. Comparison of image feature descriptor in content based image retrieval system. In:**2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)**. [S.l.: s.n.],2017. p. 1509–1513.

SOUZA, BRUNO; CARVALHO, ANDRE; SOARES, CARLOS. Metalearning for gene expression data classification. In: . [S.l.: s.n.], 2008. p. 441-446. ISBN 978-0-7695-3326-1.

SURESH, P.; SUNDARAM, R. M. D.; ARUMUGAM, A. Feature extraction in compressed domain for content based image retrieval. In:**2008 International Conference on Advanced Computer Theory and Engineering**. [S.l.: s.n.], 2008. p. 190–194