



UNIVERSIDADE ESTADUAL DE CAMPINAS
INSTITUTO DE MATEMÁTICA, ESTATÍSTICA E COMPUTAÇÃO CIENTÍFICA
DEPARTAMENTO DE MATEMÁTICA APLICADA



Vinícius Figueiredo Fernandes

Técnicas da teoria do caos em análise de imagens médicas*

Monografia apresentada ao Instituto de Matemática, Estatística e Computação Científica da Universidade Estadual de Campinas como parte dos requisitos para obtenção de créditos na disciplina Projeto Supervisionado, sob a orientação do Prof. João Batista Florindo.

Campinas
2019

*Este trabalho foi financiado pela Serra Pilheira, projeto Julho/2019.

1 Introdução ao Projeto

Com o objetivo de aumentar a acurácia e robustez no processo de classificação de imagens de texturas, foi realizado sob a orientação do Professor João Batista Florindo o estudo e implementação de diferentes algoritmos para a extração de vetores de características de imagens de texturas. A metodologia proposta foi baseada no conceito conhecido como caminhada determinística do turista. Nesta abordagem, um “agente” fictício percorre a imagem seguindo regras que dependem do nível de cinza do pixel em que o agente se encontra e de seus vizinhos. Tópicos como visão computacional e *machine learning* foram estudados para a execução deste projeto. Além da implementação e validação da caminhada determinística do turista originalmente apresentada em Backes et al.(2010), também foram testadas versões modificadas aplicadas em mapas transformados por técnicas como padrões locais binários (LBP) e Transformada de Fourier.

2 Visão Computacional

Visão computacional é um amplo campo de pesquisa que tem como objetivo a boa tomada de decisões sobre objetos físicos com base em suas imagens. [1]

A grande dificuldade nesse campo de pesquisa está em recuperar descritores de uma cena dada informação insuficiente para a total representação do problema.[4]. Um campo irmão da visão computacional é o da análise de imagens que estuda transformações e melhoramentos em imagens, porém sem o desejo de buscar interpretações sobre a cena analisada.[4] Nesse trabalho, visão computacional e análise de imagens são usados em conjunto.

A visão computacional é conhecida por ser uma área orientada à problemas[1], sendo no nosso caso, motivada pela busca de reconhecimento de padrões em imagens médicas.

Para o reconhecimento de padrões deveremos ensinar uma máquina através de amostras, para esse projeto, isso corresponde a ensinar uma máquina a caracterizar descritores após aprendizado de imagens previamente classificadas, i.e., trataremos de aprendizado de máquina supervisionado.

Para nosso modelo de classificação, precisaremos de um conjunto de m classes de objetos de maneira que cada classe corresponda a um conjunto de objetos com propriedades comuns entre si. Cada objeto será atribuído a uma classe através de um *label* ou *rótulo*. Para isso, precisaremos de um *classificador*, um algoritmo que receba uma representação de um objeto e forneça um rótulo da classe desse objeto. O processo de classificação então corresponde a atribuição de rótulos a objetos de acordo com alguma propriedade desse objeto.[1]

Um *vetor de características* ou *feature* é o vetor que representa um objeto, podendo ser a área desse objeto, seu perímetro, centroide ou qualquer outra propriedade que permita distinguir objetos semelhantes de dessemelhantes.[1]

Para a obtenção desses descritores pode ser interessante o uso de aprimoramentos ou uso de filtros em imagens. O uso desses processamentos de imagens podem permitir a melhor identificação de objetos de interesse, por exemplo, separando um objeto de interesse de seu fundo.

Assim, definimos que *processamento de imagens* é a criação de uma imagem de saída dada uma imagem de entrada e *aprimoramento de imagens* é operar de maneira que ocorram melhoramentos na imagem para a detectabilidade de detalhes importantes.[1] Dessa maneira, podemos, por exemplo, usar máscaras de modo a criar uma segunda imagem a partir da primeira que possua propriedades importantes para a extração de *features*.

Além das transformações em imagens já definidas, uma outra também importante é a *máscara* que é um conjunto de posições de pixels de uma imagem e seus correspondentes valores chamados de *pesos*. Aplicar uma máscara em uma imagem de entrada produz uma imagem de saída de mesmo tamanho da entrada.[1]

Como exemplo de máscaras, temos as máscaras para a detecção de bordas. Essas máscaras são divididas em máscaras para a detecção de bordas de primeira e segunda ordem.[3]

- Detecção de Bordas de Primeira Ordem:
Os métodos de primeira ordem buscam pelas maiores mudanças de intensidades dos pixels através da imagem, os pontos das maiores mudanças são consideradas bordas. Nesses métodos o gradiente é o vetor que exerce esse papel. Como exemplo temos a máscara de Sobel que detecta a direção de maior mudança possível da claridade da imagem e a taxa dessa mudança em cada ponto.[1]
- Detecção de Bordas de Segunda Ordem:
Os métodos de segunda ordem calcula o laplaciano de uma imagem, detectando *zero crossing edges*. *Zero crossing edge* busca por valores em que o laplaciano passa por zero, i.e., pontos em que o laplaciano muda de sinal. Como exemplo temos o LOG que detecta bordas através do Laplaciano da Gaussiana.[1]

É importante, portanto, o uso de máscaras que capturam bem as características da imagem que queremos detectar.

Um outro importante descritor das imagens que pode ajudar em sua segmentação é a textura. *Textura* é um descritor que ajuda a segmentar imagens em regiões de interesse para a classificação de tais regiões. A textura de uma imagem fornece informações da organização espacial da imagem, sendo um conjunto de objetos primitivos chamados *texels* em alguma relação de regularidade ou repetição. De outra maneira, textura pode ser interpretada como uma medida quantitativa da organização da intensidade de uma região da imagem.[1]

Para a classificação de texturas em termos de *texels* é necessário que essas regiões das imagens sejam facilmente segmentadas. Porém, como segmentar *texels* pode ser difícil em imagens reais, técnicas estatísticas para a determinação de texturas, apesar de menos intuitivas, são computacionalmente mais eficientes e portanto mais utilizadas.[1]

Para a análise de texturas, existem diversos métodos clássicos e estado-da-arte, podemos citar o LBP e os descritores de Fourier, ambos métodos utilizados durante os experimentos realizados nesse projeto. Outros métodos também poderão ser utilizados para fins comparativos.

LBP ou *Local Binary Patterns* é uma abordagem simples e robusta utilizada na classificação e segmentação de imagens.

Nesse método, a imagem é dividida em janelas de tamanhos $N \times N$. Nessa janela, os *pixels* de mesma distância ao redor de um pixel central são comparados ao valor desse, caso seja menor é atribuído o valor 0 e o valor 1 é atribuído caso contrário.

A sequência desses valores é uma sequência binária que convertida em número decimal e realizada por toda a imagem, a codifica.[5]

Nesse trabalho, trataremos o método LBP como um tipo de melhoramento da imagem que a torna invariante quanto a rotação, dessa maneira permitindo a detecção de possíveis novos descritores.

No caso do método de descritores de Fourier, os descritores são extraídos através do espectro de Fourier da imagem de textura.[5]

Como vimos, a área da visão computacional é orientada à problemas e uma etapa importante para o caso da classificação de objetos é a extração de *vetores de características* ou *features*. Desse modo, sabemos que existem diversos descritores possíveis e a escolha desses dependem da classe de problemas que estamos a avaliar. Por exemplo, para o caso do problema de identificar flores, podemos escolher descritores que bem caracterizam forma, textura e cor.

A escolha dos tipos de *vetores de características* também depende se pretendemos avaliar a imagem de maneira global ou local. Um descritor global não avalia os *pontos de interesse*, i.e., pontos chaves na imagem que possuem propriedades de interesse. Entre esses descritores temos o histograma de cor, os momentos invariantes de uma imagem que avaliam forma, o LBP que avalia textura, o HOG (Histogram of Oriented Gradients), descritor utilizado para a identificação de objetos e outros. Um método de descritor local por outro lado busca pelos pontos de interesse, extraindo descritores através dessas regiões locais ricas em informações.[10]

3 Caminhadas Determinísticas

Apesar do campo de caminhada determinística ser muito menos estudado que o de caminhadas aleatórias, a caminhada determinística tem muitas aplicações em campos como o da física, sociologia, ecologia, economia e outros. Os desafios de tal formulação matemática pode ser visto no problema clássico do caixeiro-viajante.[6]

Problemas do tipo caminhada determinística envolvem percorrer pontos com certa distribuição espacial de modo que exista uma série de regras que decida a que ponto o viajante deve ir dado o ponto em que ele se encontra e em alguns modelos, dado os pontos percorridos anteriormente. Esses critérios podem ser, por exemplo, buscar pontos com bastante recurso disponível que estejam o mais próximo possível. Esses tipos de problemas, possuem ao contrário do que se possa pensar, comportamento não-trivial, i.e., é possível produzir comportamento não-periódico e não-estacionário.[7]

Então, um dos problemas de interesse seria através de uma série de regras pré-estabelecidas, realizar buscas pelas decisões de menor custo, com recursos distribuídos aleatoriamente. Por exemplo, um sistema ecológico com recursos distribuídos desordenadamente em que o forrageamento nesse ambiente busca maximizar a quantidade de recursos coletados com o menor esforço.[6]

Nesse mesmo espírito da formulação do problema ecológico, podemos de maneira mais abstrata, pensar em andanças em um espaço abstrato de características fenotípicas. Essa formulação de um problema evolucionista considera o efeito Rainha Vermelha, uma hipótese evolucionária inspirada no livro Alice de Lewis Carroll de que os organismos devem constantemente evoluir e se adaptar.[7]

O efeito Rainha Vermelha pode ser interpretado como um modelo matemático para avaliar evolução endogênica, i.e., evolução na ausência de qualquer perturbação ambiental externa. Esses modelos estritamente determinísticos produzem comportamento não-estacionário e não-periódico, como previamente comentado. Sendo que essa aleatoriedade da caminhada é consequência direta da aleatoriedade das condições iniciais.[7]

Em outra nota, o modelo utilizado na caminhada da Rainha Vermelha de Freund e Grassberger pode ser entendido como um autômato celular com ações muito esparsas, essa constatação pode ser interessante para trabalhos futuros comparando a resolução de nossa classe de problemas com autômatos celulares em redes complexas e caminhada determinística turística que discutiremos a seguir.[7]

A caminhada determinística do turista é um tipo de caminhada *self-avoiding*, no mesmo sentido da caminhada aleatória Laplaciana, em que um viajante pode visitar N pontos distantes em um mapa d dimensional com alguma regra determinística de modo que o ponto seguinte não tenha sido visitado nos últimos μ passos. No caso da caminhada desse projeto, utilizaremos duas regras distintas, a regra do mínimo: busca pelo menor valor da intensidade dos pixels em uma vizinhança e a regra do máximo: busca pelo maior valor da intensidade dos pixels em uma vizinhança. Assim, uma única memória μ gera dois vetores de características distintos $\psi_{\mu,din}(max)$ e $\psi_{\mu,din}(min)$.

A caminhada é dividida em duas partes, uma parte transiente de tamanho t e uma parte formada por um atrator de período $p \geq \mu + 1$. O conjunto atrator indica o comportamento característico para qual o modelo evolui.

A caminhada é estritamente dependente da memória μ , do conjunto de N pontos e do ponto inicial. De modo que N pontos distintos implica em N caminhadas distintas possíveis. Assim, se definirmos a distribuição conjunta $S_{\mu,d}^N(t,p)$, cada posição da distribuição armazenará a quantidade de caminhadas de tempo transiente t e tamanho de período p . [8]

$$S_{\mu,d}^N(t,p) = \frac{1}{N} \sum_{i=1}^N \begin{cases} 1, & \text{se } t_i = t, p_i = p \\ 0, & \text{caso contrário} \end{cases} \quad (1)$$

Para a extração de informações resultantes das caminhadas construiremos um histograma $h_{\mu,din}(t+p)$ de tamanho $t+p$ definido de modo que

$$h_{\mu,din}(\mu+1) = S_{\mu,2}^N(0, \mu+1)$$

$$h_{\mu,din}(\mu+2) = S_{\mu,2}^N(0, \mu+2) + S_{\mu,2}^N(1, \mu+1)$$

$$h_{\mu,din}(\mu+3) = S_{\mu,2}^N(0, \mu+3) + S_{\mu,2}^N(1, \mu+2) + S_{\mu,2}^N(2, \mu+1)$$

Sendo possível construir um vetor de característica ou *descriptor* $\psi_{\mu,din}$ com mais informações. Definindo-o como uma concatenação dos histogramas:

$$\psi_{\mu,din} = [h_{\mu,din}(\mu+1), h_{\mu,din}(\mu+2), \dots, h_{\mu,din}(\mu+n)]. \quad (2)$$

4 Classificação Estatística

Nesta parte discutiremos brevemente as bases de imagens utilizadas e os métodos utilizados para a classificação das bases. A base Brodatz é uma base formada por 111 imagens de textura em tons de cinza. Para esse trabalho as imagens foram divididas em 111 classes de imagens com 16 imagens cada, totalizando em 1776 imagens de tamanho 128 X 128 pixels.[5] A segunda base utilizada, foi a base KTH-TIPS2 (Textures under varying Illumination, Pose and Scale) imagens coloridas de difícil classificação pela variação das texturas, cada uma das 11 classes de imagens possui 4 amostras diferentes (a,b,c,d), com diferentes posições, iluminações e escalas.[9]

Para a validação dos vetores de características das bases de textura utilizamos técnicas de aprendizagem estatísticas clássicas para classificação:

A Máquina de Vetores de Suporte ou *SVM* é um método de aprendizagem estatística em que o classificador f é um hiperplano $f(x) = (w \cdot x) + b$, em que w é chamado de *vetor-peso* e b é o *bias*, que separa os dados em um espaço N-dimensional em diferentes classes. Apesar de ser construído inicialmente para apenas separação em duas classes, pode-se estendê-lo para múltiplas classes. O método busca pelo plano ótimo que maximiza a margem, i.e., maximiza a distância entre os pontos de duas classes distintas.[5]

A Análise de Componentes Principais ou *PCA* é uma técnica utilizada para redução de dimensionalidade e portanto elimina as correlações que mascaram as informações discriminantes. Esse método gera novas variáveis em um novo espaço através de combinações lineares das variáveis antigas de modo que um número reduzido de variáveis contenham a maior parte da variabilidade nos dados, sendo suficientes para o processo de classificação.[5]

A Análise de Discriminantes Lineares ou *LDA* é uma técnica de transformação linear que também pode ser usada para redução de dimensionalidade, porém, ao contrário do PCA, LDA é uma técnica supervisionada. LDA tenta maximizar a razão entre a variância de classes distintas e a variância dentro da mesma classe. A ideia é buscar por um vetor de projeção a que pode ser usado para computar projeções escalares para cada vetor de entrada x , de modo que, $y = ax$. [11]

5 Experimentos e Resultados

Os programas foram implementados na linguagem Python 3.7 utilizando-se também dos pacotes numpy, pillow e scikit-image instalados no ambiente Spyder 3.3.3 no GUI (*graphical user interface*) Anaconda Navigator.

Para o algoritmo clássico da caminhada determinística do turista, i.e., sem o uso de LBP, na base Brodatz foram obtidos os seguintes resultados utilizando LDA:

Memória utilizada(μ)	Max e/ou Min	Porcentagem de acerto
0	Max	45,0%
1	Max	47,4%
2	Max	46,2%
3	Max	42,2%
0'	Min	36,8%
1'	Min	30,2%
0 e 1	Max	58,5%
0 e 1 e 2	Max	69,3%
0 e 1 e 2 e 0'	Max e Min	78,8%

Para o algoritmo da caminhada determinística utilizando de LBP de raio 2 e com 12 pontos na base Brodatz foram obtidos os seguintes resultados utilizando LDA: Além disso, misturando vetores de características do caso clássico e LBP, obtivemos:

Memória utilizada(μ)	Max e/ou Min	Porcentagem de acerto
0	Max	32,5%
1	Max	31,8%
2	Max	35,0%
0'	Min	35,7%
0 e 1	Max	62%
0 e 1 e 2	Max	68,6%
0 e 1 e 2 e 0'	Max e Min	74,3%

Memória utilizada(μ)	Max e/ou Min	Porcentagem de acerto
0 e 1 e 2 e 0' e 0(clássico)	Max e Min	85,0%
0 e 1 e 2 e 0' e 0(clássico) e 0'(clássico)	Max e Min	87,0%

Como teste prático o algoritmo da caminhada determinística do tipo normal, i.e., sem modificação LBP e com apenas uma memória ($\mu = 0$ para máximo) foi aplicado em uma base de imagens de cistos mandibulares (OKC). Essa base consiste em 3 classes de imagens: K (cisto esporádico), R (cisto radicular) e S (cisto sindrômico). A classe K possuindo 1300 imagens, a classe R, 900 imagens e a classe S, 800 imagens. Fazendo a discriminação das 3 classes (k x s x r) foi encontrado 54,2% de acerto.[13]

6 Conclusão

Mesmo com os avanços na tecnologia *Deep*, é possível com técnicas clássicas para a extração de vetores de característica para algumas classes de imagens conseguir resultados concorrentes. Além do mais, como existem métodos mistos para a detecção de objetos em imagens, que se utilizam de técnicas clássicas e *Deep*, fica evidente que técnicas clássicas não devem ser totalmente descartadas, pois seu estudo ainda apresenta resultados interessantes.

Referências

- [1] SHAPIRO, L. G., STOCKMAN, G. C. Computer vision. Upper Saddle River, NJ: Prentice Hall. 2001.
- [2] BEZDEK, J. What is Computational Intelligence?. Computational Intelligence: Imitating Life, J.M Zurada, R.J Marks II and C.J Robinson (Eds) IEEE Press. 1-12, 1994.
- [3] Alazzawi, Abdulbasit, Alsaadi, Husam, Shallal, Abidaoun, Albawi, Saad. EDGE DETECTION-APPLICATION OF (FIRST AND SECOND) ORDER DERIVATIVE IN IMAGE PROCESSING. 2015.
- [4] SZELISKI, Richard. Computer Vision: Algorithms and Applications (1st ed.). Springer-Verlag, Berlin, Heidelberg. 2010.
- [5] FLORINDO, João Batista. Descritores fractais aplicados à análise de texturas. 2013. Tese (Doutorado em Física Aplicada) - Instituto de Física de São Carlos, Universidade de São Paulo, São Carlos, 2013. doi:10.11606/T.76.2013.tde-02052013-161100. Acesso em: 2019-10-20.
- [6] BOYER, Denis, LARRALDE, Hernán. Looking for the right thing at the right place: Phase transition in an agent model with heterogeneous spatial resources. Complexity. 2005. 10. 52-55. 10.1002/cplx.20068.
- [7] FREUND, H., GRASSBERGER, P. The red queen's walk. Phys. A Stat. Mech. Appl., 190 (1992), pp. 218-237, 10.1016/0378-4371(92)90033-M Acesso em: 2019-10-25.
- [8] GONÇALVES, Wesley Nunes. Caminhadas determinísticas em redes complexas aplicadas em visão computacional. 2010. Dissertação (Mestrado em Ciências de Computação e Matemática Computacional) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2010. doi:10.11606/D.55.2010.tde-08042010-112016. Acesso em: 2019-10-24.
- [9] The KTH-TIPS and KTH-TIPS2 image databases.
Disponível em: <https://www.nada.kth.se/cvap/databases/kth-tips/>.
Acesso em: 20 de Outubro de 2019.
- [10] Image Classification using Python and Scikit-learn. Gogul Ilango, 2019. Disponível em: <https://gogul.dev/software/image-classification-python/>.
Acesso em: 22 de Outubro de 2019.
- [11] WITTEN, Ian H., EIBE, Frank, HALL, Mark A., PAL, Christopher J., Chapter 8 - Data transformations, Editor(s): Ian H. Witten, Eibe Frank, Mark A. Hall, Christopher J. Pal, Data Mining (Fourth Edition), Morgan Kaufmann, 2017, Pages 285-334, ISBN 9780128042915, <https://doi.org/10.1016/B978-0-12-804291-5.00008-8>.
- [12] BACKES, Andre R., GONÇALVES, Wesley N., MARTINEZ, Alexandre S., MARTINEZ, Odemir B. (2010). Texture analysis and classification using deterministic tourist walk. Pattern Recognition. 43. 685-694. 10.1016/j.patcog.2009.07.017.
- [13] Florindo, João; Bruno, Odemir; Landini, Gabriel. (2016). Morphological Classification of Odontogenic Keratocysts Using Bouligand-Minkowski Fractal Descriptors. Computers in Biology and Medicine. 81. 10.1016/j.compbiomed.2016.12.003.