

Classificação de espécies de corais.

Anna Caroline Bozzi
Departamento de Informática

Universidade Federal do Paraná

Curitiba-PR, Brasil
annac.bozzi@gmail.com

Resumo— O objetivo desse projeto é classificar diferentes fotos de corais em suas respectivas espécies, através de uma base de imagens disponibilizadas pelo Laboratório de Ecologia Marinha(LECOM) da Universidade Federal do Rio Grande do Norte juntamente com fotos populares postadas no Instagram com a hashtag #DeOlhoNosCorais. Todas as imagens foram previamente rotuladas e tratadas para que as fotos fossem exclusivamente de corais de forma que fosse possível extrair textura como principal característica de classificação. Dentre os classificadores utilizados o Ensemble utilizando método Bagging de KNeighborsClassifier estimadores da base foi o que retornou melhores resultados

Palavras-chave— aprendizagem de máquina, machine learning, aprendizagem supervisionada, corais, classificação,

I. INTRODUÇÃO

Em parceria com a equipe do Laboratório de Ecologia Marinha (LECOM) e o pesquisador Guilherme Ortega Longo da Universidade Federal do Rio Grande do Norte, esse trabalho propõem uma classificação das espécies de corais utilizando as fotos populares postadas no Instagram com a hashtag #DeOlhoNosCorais e as fornecidas pelo LECOM. Essas fotos foram previamente rotuladas pela equipe do laboratório com a ferramenta LabelMe. através de algoritmos de classificação. O banco fornecido possuía 1510 imagens separadas em 21 classes.

O trabalho consistiu em (II-A) seleção das cinco classes com maior número de imagens, (II-B) corte para obter apenas o coral das imagens, rotuladas com a ferramenta LabelMe, (II-C) divisão das imagens de coral em pequenos patches 100x100, (II-D) limpeza e separação da base de dados dos cortes, (II-E) extração de características de textura, (II-F) classificação, (II-G) e *cross validation*.

II. METODOLOGIA

A. Seleção das cinco classes com maior número de imagens:

Nessa primeira etapa foram escolhidas as 5 classes com maior número de imagens.

TABLE I: Banco de imagens

Classe	Número de imagens
Agracia spp	73
Favia Gravida	63
Milepora Alaicornis	150
Montastrea Cavernosa	118
Palythoa Caribaeorum	105

B. Corte apenas do coral das imagens previamente rotuladas com a ferramenta LabelMe:

A rotulação aconteceu através da ferramenta LabelMe, com ela é possível contornar o coral na foto e salvando essas

coordenadas em um arquivo JSON. Em seguida foi feito um script que processa as coordenadas do JSON de forma a realizar o corte, Figura 1.

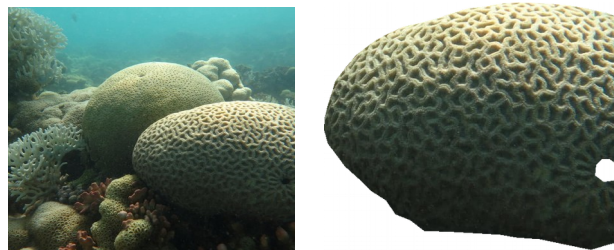


Fig. 1: Exemplo de corte do coral.

C. Divisão das imagens cortadas em pequenos patches de 100x100:

Tendo a imagem do coral previamente cortada em seguida foi dividida essa imagem em pequenos patches de tamanhos 100x100, Figura 2, para que fosse passado para os classificadores, as respectivas característica das imagens padronizadas.

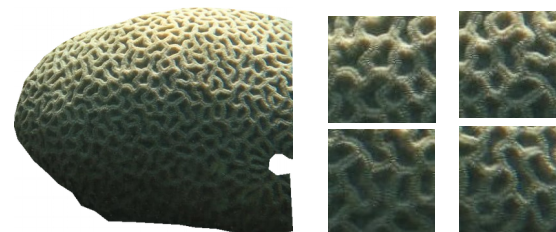


Fig 2: Exemplo de divisão em patches.

D. Limpeza e separação da base de imagens de patches:

Cada imagem gerava um número variável de patches, foram então removidas as imagens que possuíam outras informações além do coral na foto, e também as com qualidade inferior para que os bancos de cada classe ficassem balanceados.

Em seguida foram divididos as cinco classes de forma aleatória em 3 bancos balanceados, os patches da mesma imagem permaneceram no mesmo banco.

TABLE II : Bancos de patches

Classe	Banco 1	Banco 2	Banco 3
Agracia spp	1000	924	818
Favia Gravida	1000	980	1000
Milepora Alaicornis	964	1000	1000
Montastrea Cavernosa	1000	1000	970
Palythoa Caribaeorum	1000	903	1000

E. Extração de características de textura:

Para a extração das características de textura das imagens, patches, foi usado LBP, ele resume as estruturas locais na imagem gerando uma representação estatísticas dessas estruturas. Quando os pixels ao redor são todos pretos ou brancos, a região da imagem é plana, ou seja, sem características. Grupos de pixel pretos ou brancos contínuos são padrões “uniformes”, interpretados como cantos ou arestas, se os pixels alternarem entre pretos e brancos o padrão é considerado “não uniforme”, Figura 3.

Foi utilizado a função *local_binary_pattern* da biblioteca *skimage.feature* em seguida normalizado o histograma retornado, e por fim passado para o arquivo de característica juntamente com a sua respectiva classe.

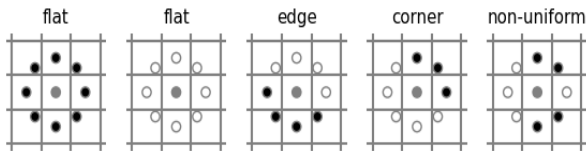


Fig 3. Representação do LBP.

F. Classificação:

Para a classificação foi utilizados:

- *Support Vector Machine*;
- *Decision Tree*;
- *Ensamble(Bagging(KNeighborsClassifier))*;

Para cada banco previamente separado foram rodados cada um dos classificadores, utilizando 70% da base para treinamento e 30% para teste, e foi reportado acurácia e a matriz de confusão de cada classificador para análise.

G. Cross validation:

Para avaliação final a fim de conclusões foi utilizado a etapa de cross validation para os 3 classificadores, intercalando os treinamentos e testes em:

- Treino: Banco 1+ Banco 2
Teste: Banco 3
- Treino: Banco 1+ Banco3
Teste: Banco 2
- Treino: Banco 2 + Banco 3
Teste: Banco 1

III. RESULTADOS

A tabela III apresenta a acurácia dos 3 classificadores utilizados para cada banco individualmente.

TABLE III : Acurácia de cada classificador em relação ao banco

Classificador	Banco 1	Banco 2	Banco 3
<i>Support Vector Machine</i>	0.7778	0.7498	0.6613
<i>Decision Tree</i>	0.6758	0.6791	0.6765
<i>Ensamble</i>	0.8087	0.7893	0.7672

O Ensamble(Bagging(KNeighborsClassifier)) foi o que resultou em melhor acurácia em relação aos outros, e é possível observar que a média de acurácia entre os classificadores é melhor no Banco 1.

A seguir na tabela IV segue os resultados da acurácia da aplicação do cross validation em cada classificador.

TABLE IV: Resultados *cross validation*

Classificador	Acurácias
<i>Support Vector Machine</i>	0.7728 0.7830 0.7212
<i>Decision Tree</i>	0.6773 0.6779 0.6717
<i>Ensamble</i>	0.7830 0.7679 0.7424

IV. CONCLUSÃO

Os três classificadores demonstraram bons resultados gerais em relação aos testes realizados. O Ensamble foi o que melhor se desempenhou, manteve uma média de 0.7644 na acurácia em cross validation.

A fim de novos experimentos podem ser avaliados diferentes tamanhos de patches nesses classificadores, bem como novos extratores de características das imagens e também ainda para diferentes tamanhos de bancos e balanceamentos.

REFERENCES

- [1] Estéfano van der Walt, Johannes L. Schönberger, Juan Nunez-Iglesias, François Boulogne, Joshua D. Warner, Neil Yager, Emmanuelle Gouillart, Tony Yu, and the scikit-image contributors. *scikit-image: Image processing in Python*. *PeerJ* 2:e453 (2014) <https://doi.org/10.7717/peerj.453>
- [2] LabelMe: a database and web-based tool for image annotation . B. Russell, A. Torralba, K. Murphy, W. T. Freeman. *International Journal of Computer Vision*, 2007.
- [3] Scikit-learn: Machine Learning in Python, Pedregosa et al., *JMLR* 12, pp. 2825-2830, 2011.

