

Classificação de Imagens no Conjunto de Dados MPEG7

Alexandre Rafael Rodrigues, Filipe Ferreira Viana, Pedro Lucas Alves de Carvalho

Universidade Federal de Viçosa

Rio Paranaíba, MG, Brasil

alexandre.rafael@ufv.br, filipe.f.viana@ufv.br, pedro.l.carvalho@ufv.br

Abstract—Este trabalho apresenta um *pipeline* completo para classificação de imagens do conjunto MPEG7 modificado. As etapas incluem segmentação, extração de características morfológicas, normalização e classificação utilizando os métodos k-NN e Random Forest. Resultados detalhados mostram que o Random Forest apresentou uma acurácia de 96,67%, superando o k-NN com 94,17%. Além disso, são discutidas as principais dificuldades, erros e possíveis melhorias, como a inclusão de redes neurais em futuros trabalhos.

Index Terms—Classificação de Imagens, MPEG7, k-NN, Random Forest, Visão Computacional.

I. INTRODUÇÃO

A classificação de imagens é uma tarefa fundamental na visão computacional, com aplicações que abrangem diversas áreas, como saúde (diagnósticos médicos baseados em imagens), segurança (reconhecimento facial e detecção de objetos) e indústria (controle de qualidade e automação). A capacidade de categorizar imagens de forma precisa e eficiente é essencial para desenvolver sistemas inteligentes que possam interpretar e tomar decisões com base em dados visuais.

Neste trabalho, abordamos a classificação de formas geométricas presentes no conjunto de dados MPEG7, conhecido por sua diversidade de imagens. Para realizar a tarefa de classificação, dois métodos de aprendizado supervisionado foram explorados: o *k-Nearest Neighbors* (k-NN), escolhido por ser um modelo amplamente reconhecido na resolução de problemas de classificação, devido à sua simplicidade e eficiência em contextos menos complexos, e o *Random Forest*, selecionado por ser um modelo mais robusto e complexo, capaz de capturar relações não lineares e generalizar melhor em cenários mais desafiadores.

O objetivo principal deste trabalho é implementar um *pipeline* completo para classificação de imagens, que inclua segmentação, extração de características, normalização e classificação, e avaliar o desempenho dos dois modelos mencionados. Métricas como acurácia, precisão, recall e F1-score foram utilizadas para comparar os modelos, destacando suas forças e limitações.

II. METODOLOGIA

A metodologia segue uma estrutura que contém cinco etapas principais: segmentação, extração de características, divisão de dados, normalização e classificação.

A. Segmentação de Imagens

Os dados fornecidos já estavam previamente segmentados, eliminando a necessidade de realizar o processo de segmentação manualmente. As imagens estavam organizadas em classes específicas, prontas para análise e extração de características.

B. Extração de Características

A extração de características morfológicas foi realizada utilizando as bibliotecas `pandas` e `cv2` (OpenCV), que são amplamente reconhecidas no processamento e manipulação de dados em visão computacional. A biblioteca `cv2` foi utilizada para operações como leitura de imagens, binarização e cálculo de propriedades geométricas, enquanto a `pandas` foi empregada para armazenar e organizar as informações extraídas em um formato tabular estruturado (`DataFrame`).

As principais características morfológicas extraídas foram a área, que representa o número total de pixels dentro do contorno da forma; o perímetro, que corresponde ao comprimento total da borda da forma; a circularidade, que indica o quão similar a forma é a um círculo; a razão de aspecto, que mede a relação entre largura e altura da forma, sendo útil para diferenciar formas alongadas de mais simétricas; e a excentricidade, que reflete o desvio da forma em relação a um círculo perfeito, evidenciando irregularidades ou alongamentos.

C. Divisão e Normalização

Os dados foram organizados em três subconjuntos: treino (70%), validação (15%) e teste (15%). Essa divisão foi feita utilizando a função `train_test_split` da biblioteca `scikit-learn`, que possibilita uma divisão aleatória e controlada dos dados. Para garantir reprodutibilidade, foi utilizada uma `seed` fixa (42) no processo de randomização.

Antes do treinamento, as características foram normalizadas para a faixa $[0, 1]$ usando o método Min-Max. Essa normalização é importante para evitar que características com valores maiores dominem os modelos, garantindo que todas as características tenham a mesma influência no treinamento dos classificadores.

D. Classificação

Dois classificadores foram implementados e avaliados:

- **k-Nearest Neighbors (k-NN):** Modelo baseado em vizinhos mais próximos, onde a classe de um exemplo é definida pela maioria das classes de seus k vizinhos mais próximos no espaço de características. Após validação, o valor $k = 3$ foi escolhido como o mais adequado, mostrando um desempenho melhor em classes com formas regulares e bem separadas.
- **Random Forest:** Modelo baseado em *ensembles*, que combina múltiplas árvores de decisão para obter maior eficácia e capacidade preditiva. Foi configurado com 100 árvores ($n_estimators=100$) e profundidade ilimitada ($max_depth=None$), permitindo capturar padrões complexos e relações não lineares nos dados.

III. RESULTADOS E DISCUSSÃO

Os resultados obtidos no conjunto de teste estão resumidos na Tabela I. As matrizes de confusão são apresentadas nas Figuras 1 e 2.

TABLE I
DESEMPENHO NO CONJUNTO DE TESTE

| Métrica | k-NN | Random Forest |
|----------------|--------|---------------|
| Acurácia | 94,17% | 96,67% |
| Precisão Média | 94% | 97% |
| Recall Médio | 94% | 97% |
| F1-Score Médio | 94% | 97% |

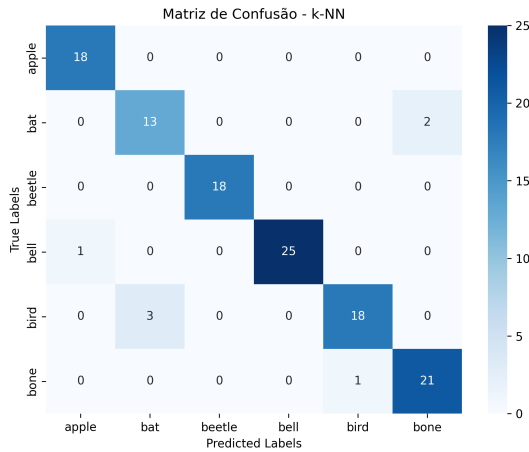


Fig. 1. Matriz de Confusão - k-NN

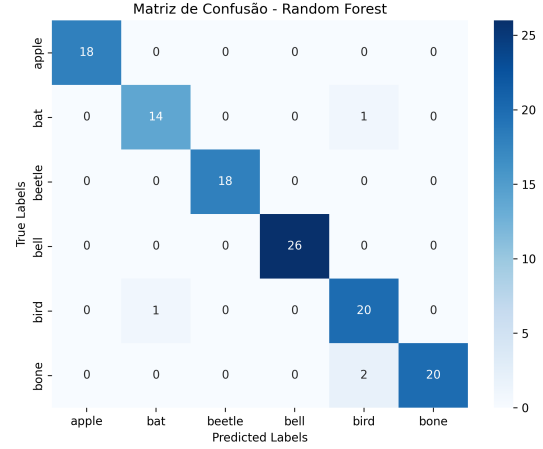


Fig. 2. Matriz de Confusão - Random Forest

O Random Forest superou o k-NN em todas as métricas, mostrando sua capacidade de capturar relações mais complexas nos dados. As principais confusões do k-NN ocorreram entre as classes *bat* e *bird*, que possuem alta similaridade visual. As confusões mais frequentes foram observadas nas classes *bat* e *bird* para o k-NN, enquanto o Random Forest apresentou erros residuais na classe *bone*. Isso sugere que características adicionais, como textura ou momentos geométricos, poderiam ser exploradas para melhorar o desempenho.

IV. CONCLUSÃO

Este trabalho implementou um *pipeline* para classificação de imagens no conjunto MPEG7. O *Random Forest* destacou-se com acurácia de 96,67%, superando o *k-NN* 94,17%.