

Classificação de Imagens no Conjunto de Dados MPEG7 Modificado

1st Julio Cezar Lopes Cardoso

Universidade Federal de Viçosa

Campus Rio Paranaíba

Minas Gerais, Brasil

Juliocezarlopescardoso@gmail.com

2nd Gustavo Henrique de Deus Reis

Universidade Federal de Viçosa

Campus Rio Paranaíba

Minas Gerais, Brasil

gustavo.reis@ufv.br

Abstract—In this project, the objective is to train and test a classifier for the modified MPEG7 dataset. The project steps include image segmentation, degradation of morphological features, data normalization and classifier performance analysis.

I. INTRODUÇÃO

Neste projeto, o objetivo é treinar e testar um classificador para o conjunto de dados MPEG7 modificado. As etapas do projeto incluem segmentação das imagens, degradação de características morfológicas, normalização dos dados e análise de desempenho do classificador.

II. SEGMENTAÇÃO DAS IMAGENS

A etapa de segmentação das imagens teve como objetivo identificar e isolar as formas presentes em cada imagem do conjunto de dados. Para isso, utilizamos as seguintes técnicas:

A. Limiarização (Thresholding):

Aplicamos uma transformação binária às imagens em tons de cinza para separar os pixels de interesse (objetos) do fundo. A técnica utilizada foi o thresholding global, com um limiar de valor fixo igual a 127. Os pixels com valores superiores ao limiar foram marcados como brancos (255), enquanto os inferiores foram marcados como pretos (0).

B. Detecção de Contornos:

Após a binarização, os contornos das formas foram identificados utilizando a função 'cv2.findContours' do OpenCV. Os contornos externos das formas foram selecionados para garantir que cada imagem fosse representada por um conjunto único de bordas.

C. Resultado:

Para cada imagem, foi gerado um conjunto de contornos que representavam as formas segmentadas. Essas formas foram então processadas na etapa seguinte para extração de características morfológicas.

Essa etapa garantiu uma separação precisa dos objetos de interesse no conjunto de dados, permitindo que as etapas posteriores de extração de características e classificação fossem realizadas com base em dados bem estruturados.

III. EXTRAÇÃO DE CARACTERÍSTICAS MORFOLÓGICAS:

A etapa de extração de características morfológicas foi realizada para transformar as formas segmentadas em dados quantitativos, facilitando a análise e a classificação das imagens. As seguintes características foram calculadas para cada forma:

Área: Representa o número total de pixels dentro do contorno detectado. Calculada utilizando a função `cv2.contourArea` do OpenCV. Perímetro: Mede a soma dos comprimentos de todos os lados do contorno. Calculada utilizando a função `cv2.arcLength`, considerando que o contorno é fechado. Circularidade:

Avalia a forma do objeto com base na relação entre área e perímetro. Fórmula utilizada:

A circularidade de uma forma é definida pela seguinte fórmula:

$$\text{Circularidade} = \frac{4\pi \cdot \text{Área}}{\text{Perímetro}^2}$$

Onde:

- Área: Número total de pixels dentro do contorno da forma.
- Perímetro: Comprimento total do contorno da forma.
- Valores de Circularidade próximos de 1 indicam formas mais circulares.

Com valores próximos a 1 indicando formas mais circulares. Um perímetro igual a zero foi tratado para evitar erros de divisão. Resultado:

Para cada forma detectada, as características acima foram computadas e armazenadas em uma tabela, juntamente com o rótulo correspondente (classe da imagem). Essas características foram utilizadas como base para as etapas de normalização e classificação. Essa etapa permitiu transformar as formas segmentadas em dados quantitativos, fornecendo informações úteis para a construção de modelos de aprendizado de máquina.

A tabela a seguir apresenta as características morfológicas extraídas do conjunto de dados, incluindo área, perímetro e circularidade, juntamente com os rótulos das classes:

Área	Perímetro	Circularidade	Rótulo
25442.0	757.47	0.55	bell
2499.0	240.30	0.54	bell
5266.0	311.13	0.68	bell
25082.5	837.91	0.44	bell
25070.5	802.70	0.48	bell

TABLE I
CARACTERÍSTICAS MORFOLÓGICAS EXTRAÍDAS

IV. DIVISÃO DO CONJUNTO DE DADOS

Para realizar a divisão do conjunto de dados, seguimos as diretrizes estabelecidas:

A. Divisão Inicial:

O conjunto de dados foi dividido em dois subconjuntos principais: *treino* + *validação* e *teste*, sendo que o conjunto de teste foi definido como **30% do total** de amostras. Essa divisão foi realizada utilizando a função `train_test_split` da biblioteca Scikit-learn, com a **seed 42** para garantir reprodutibilidade.

B. Divisão Secundária:

O subconjunto de *treino* + *validação* foi dividido novamente em dois subconjuntos: **treino** e **validação**. O tamanho do conjunto de validação foi definido como **20% do conjunto de treino + validação**.

C. Resultados:

A divisão final ficou da seguinte forma:

Treino = 336 amostras
Validação = 84 amostras
Teste = 180 amostras

D. Justificativa:

- A divisão dos dados visa garantir que os classificadores sejam treinados em um conjunto amplo e avaliados em um conjunto independente, com uma parte reservada para ajustar hiperparâmetros (*validação*).

Essa abordagem garante um processo de aprendizado de máquina robusto e permite a avaliação confiável dos modelos no conjunto de teste.

V. NORMALIZAÇÃO DOS DADOS

A. Motivação

As características morfológicas, como área, perímetro e circularidade, possuem escalas diferentes. Por exemplo, a área pode ter valores muito maiores que a circularidade. Sem normalização, características com valores maiores poderiam dominar o aprendizado, prejudicando o desempenho do modelo.

B. Técnica Utilizada

Utilizamos o `MinMaxScaler` da biblioteca Scikit-learn para normalizar os dados em um intervalo fixo de $[0, 1]$. A normalização foi ajustada no conjunto de **treino** e aplicada ao conjunto de **validação** e **teste** para evitar vazamento de dados.

C. Resultados

Após a normalização: Os dados foram reescalados para o intervalo $[0, 1]$ no conjunto de treino. Os conjuntos de validação e teste foram transformados de forma consistente com o escalador ajustado no treino.

D. Implementação

A implementação foi realizada em Python utilizando o seguinte fluxo: Ajuste do escalador no conjunto de treino: `scaler.fit(X_train)`. Transformação dos conjuntos de treino, validação e teste: `scaler.transform()`.

E. Impacto

A normalização reduziu o impacto de escalas diferentes entre as características. Garantiu que todas as características contribuíssem de forma equilibrada no aprendizado dos modelos.

Essa etapa é essencial para melhorar a estabilidade e o desempenho de algoritmos baseados em distância, como *k-NN*, e modelos baseados em gradiente.

VI. TREINAMENTO E TESTE DO CLASSIFICADOR

Modelos Selecionados

Dois classificadores foram selecionados para o treinamento e teste: **k-Nearest Neighbors (k-NN)**: Um classificador baseado na proximidade entre os pontos. **Random Forest**: Um modelo baseado em árvores de decisão que combina resultados de múltiplas árvores para aumentar a robustez.

Configuração dos Modelos

- **k-NN**:
 - Número de vizinhos (k) definido como 5.
 - Algoritmo de distância utilizado: Euclidiana.
- **Random Forest**:
 - Número de árvores na floresta ($n_estimators$) definido como 100.
 - *Seed* para reprodutibilidade: 42.

Treinamento dos Modelos

Os modelos foram treinados utilizando o conjunto de **treino**, previamente normalizado. Cada modelo foi ajustado para prever as classes com base nas características morfológicas extraídas.

Teste e Avaliação

Os modelos foram avaliados no conjunto de **teste**, com os seguintes resultados:

k-NN: **Acurácia: 91%. Matriz de Confusão:**

29	0	0	0	0	0
0	19	0	0	5	1
0	0	24	0	0	0
4	0	0	34	0	0
0	3	0	0	28	0
0	0	0	0	4	29

Relatório de Classificação: Métricas de precisão, recall e F1-score para cada classe foram calculadas e indicaram um bom desempenho geral.

Random Forest: Acurácia: 89%. Matriz de Confusão:

27	0	0	2	0	0
0	20	0	0	4	1
0	0	24	0	0	0
2	0	0	36	0	0
0	2	0	0	29	0
0	2	3	0	4	24

Relatório de Classificação: O modelo apresentou desempenho ligeiramente inferior ao k-NN, com algumas dificuldades em algumas classes.

Conclusão

Os dois modelos foram capazes de realizar uma classificação eficiente das imagens segmentadas. O modelo **k-NN** apresentou uma leve superioridade em termos de acurácia geral, destacando-se como a melhor escolha para este conjunto de dados.

VII. AVALIAÇÃO DO DESEMPENHO

A avaliação do desempenho dos classificadores foi realizada com base nos seguintes critérios:

Relatório de Classificação

Os classificadores foram avaliados com base nas seguintes métricas:

- **Acurácia:** Percentual de previsões corretas no conjunto de teste.
- **Precisão (Precision):** Proporção de previsões corretas entre as amostras classificadas como pertencentes a uma classe.
- **Revocação (Recall):** Proporção de amostras de uma classe corretamente identificadas.
- **F1-Score:** Média harmônica entre precisão e revocação, oferecendo uma visão balanceada do desempenho.

Resultados para k-NN:

- **Acurácia:** 91%.
- As métricas por classe indicaram um desempenho equilibrado, com classes como "apple" e "bone" obtendo os melhores resultados, destacando-se como a melhor opção para o conjunto de dados.

Resultados para Random Forest:

- **Acurácia:** 89%.
- O modelo Random Forest obteve precisão e revocação inferior em comparação ao k-NN.

Conclusão

A avaliação dos classificadores demonstrou que ambos os modelos apresentaram um bom desempenho, sendo o **k-NN** ligeiramente superior. O uso de métricas como matriz de confusão, precisão, revocação e F1-score foi essencial para entender a eficácia dos modelos e identificar as áreas de melhoria.

VIII. RESUMO DOS RESULTADOS

O projeto teve como objetivo a classificação de imagens segmentadas com base em características morfológicas extraídas. Abaixo, apresentamos os principais resultados:

- **Segmentação das Imagens:** As formas presentes nas imagens foram segmentadas utilizando técnicas de thresholding e detecção de contornos, garantindo a separação precisa dos objetos de interesse.
- **Extração de Características Morfológicas:** Foram calculadas características como área, perímetro e circularidade para cada forma segmentada. Esses dados serviram como base para os modelos de classificação.
- **Divisão do Conjunto de Dados:** O conjunto de dados foi dividido em:
 - Treino: 336 amostras.
 - Validação: 84 amostras.
 - Teste: 180 amostras.
- **Normalização dos Dados:** As características foram normalizadas para o intervalo $[0, 1]$, garantindo escalas uniformes para melhorar o desempenho dos modelos.
- **Treinamento e Avaliação dos Modelos:** Dois classificadores foram utilizados:
 - **k-Nearest Neighbors (k-NN):**
 - * Acurácia: 91%.
 - * Este modelo apresentou métricas ligeiramente superiores, destacando-se como a melhor escolha para este conjunto de dados.
 - **Random Forest:**
 - * Acurácia: 89%.
 - * O modelo teve bom desempenho geral, com algumas dificuldades em classes específicas.

Os resultados obtidos demonstram que as técnicas aplicadas foram eficazes para a classificação de imagens segmentadas, com o modelo k-NN obtendo o melhor desempenho geral.