

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

**PEDRO HENRIQUE DE OLIVEIRA COSTA
FELIPE MARTINS
HERBERT CARDOZO**

CLASSIFICADORES DOS PERSONAGENS DO SIMPSONS

**CIDADE
ANO**

LISTA DE FIGURAS

Figura 1 – Matriz de Confusão Árvore de Decisão	10
Figura 2 – Matriz de Confusão do <i>ensemble</i>	10
Figura 3 – Matriz de Confusão KNN	10
Figura 4 – Matriz de Confusão MLP	10
Figura 5 – Matriz de Confusão <i>Random Forest</i>	10
Figura 6 – Matriz de Confusão SVM	10

LISTA DE TABELAS

Tabela 1 – Características cromáticas dos Simpsons	5
Tabela 2 – Tabela de Hiperparâmetros	8
Tabela 3 – Tabela Resultado Validação Cruzada	9
Tabela 4 – Tabela Resultado Teste Final	9

SUMÁRIO

1	INTRODUÇÃO	4
2	MATERIAIS	5
2.1	Extração de Características	5
2.1.1	Pré-processamento das imagens	5
2.1.2	Histogramas RGB Normalizados	5
2.1.2.1	16 bins por canal e normalização	6
2.1.3	Descriptor Vermelho Verde e Azul <i>RED GREEN and BLUE</i> (RGB)	6
2.1.4	Relação entre as Características e os Resultados Obtidos	6
2.2	Classificadores Avaliados	6
2.2.1	Análise Comparativa dos Classificadores	7
2.2.2	Otimização de Hiperparâmetros e Validação Cruzada	8
3	RESULTADO	9
4	CONCLUSÃO	11
4.1	Trabalhos Futuros	11

1 INTRODUÇÃO

A tarefa de classificação de imagens tem adquirido relevância crescente nas últimas décadas, impulsionada pelo aumento do poder computacional e pela disponibilidade de bases de dados. Em particular, a classificação de personagens animados apresenta desafios específicos, como variações estilísticas, presença de ruído visual, iluminação não padronizada e pequenas diferenças morfológicas entre classes.

Este trabalho investiga técnicas clássicas de aprendizado de máquina aplicadas à classificação de cinco personagens da série animada *The Simpsons*: **Bart, Homer, Lisa, Maggie e Marge**. Para isso utilizamos a base de dados disponibilizada pelo professor, possui imagens para treino e teste, com uma quantidade limitada de exemplos por classe, aproximando-se de cenários reais de bases pequenas (problema conhecido como *small datasets*).

Para decidirmos quais características utilizar, e de qual maneira extrai-las, realizamos alguns testes, treinando e medindo a acurácia conseguida utilizando diferentes grupos de características, para assim decidirmos qual utilizar na versão final do código desenvolvido. A característica utilizada foi o Histograma RGB.

Os modelos que foram utilizados na aplicação foram os seguintes:

- **KNN** com tuning de k e pesos
- **SVM** com kernel RBF
- *Arvore de Decisão*
- **Random Forest**
- **MLP** (rede neural clássica)
- **Ensemble Voting** com 20 classificadores heterogêneos

A avaliação segue as exigências acadêmicas: **validação cruzada estratificada k=10**, métricas macro e matrizes de confusão, além de um conjunto de treino separado, garantindo avaliação robusta.

O trabalho realizado demonstra que utilizando somente o Histograma RGB é possível conseguir resultados interessantes para a classificação de qual personagem está na imagem. E o modelo que conseguiu o melhor resultado com a estratégia abordada foi o *Random Forest*.

2 MATERIAIS

2.1 Extração de Características

As características utilizadas para treinar os modelos foram escolhidas da seguinte maneira, por meio de pesquisa, conversa com alunos e professores e experiencias previas em atividades na faculdade, levantamos 3 tipos de características que poderiam ser uteis para o treinamento dos modelos, estas foram: Histograma em RGB, em *Matiz Saturação e Intensidade Hue Saturation and Intensity* (HSI) e o *Histogram of Oriented Gradients* (HOG), mas ao realizarmos os teste obtivemos que o HOG ruins, e o HSI era levemente inferior ao RGB, testamos também utilizando o RGB em conjunto com os outros dois, mas também obtivemos resultados ruins, então concluímos que utilização somente da característica do histograma RGB era nossa melhor opção.

Depois de determinarmos que o histograma RGB era a melhor característica para este problema, testamos mais de um metodo de sua utilização, acabamos utilizando o histograma em 16 *bins* por canal de cor, ou seja, conseguimos 48 características para cada instancia.

2.1.1 Pré-processamento das imagens

Antes da extração dos descritores, cada imagem passa pelo seguinte processo: **Conversão para RGB**, o uso do formato RGB padroniza os canais cromáticos, uma vez que algumas imagens podem estar em tons de cinza ou paletas indexadas; **Redimensionamento para 96×96 pixels** Definido em **IMG_SIZE = (96, 96)**, esse tamanho reduzido permite a diminuição significativa do custo computacional, normalização da dimensão entre classes, preservação suficiente dos detalhes faciais e do contorno dos personagens; **Conversão para array NumPy** Necessário para cálculos de histogramas.

2.1.2 Histogramas RGB Normalizados

A extração de informações sobre cor são essenciais neste problema, esse é o motivo da utilização dos histogramas RGB pois os personagens dos Simpsons possuem uma identidade cromática forte:

Tabela 1 – Características cromáticas dos Simpsons

Personagem	Características cromáticas
Bart	camiseta vermelha, pele amarela vibrante, cabelo amarelo irregular
Homer	camisa branca, barba azul clara
Marge	cabelo azul alto, vestido verde
Lisa	vestido vermelho, colar branco
Maggie	roupa azul-bebê, laço azul

Essas cores são informação altamente discriminativa, especialmente considerando que as imagens foram redimensionadas para uma baixa resolução (96×96).

2.1.2.1 16 bins por canal e normalização

Um histograma de 16 bins por canal é um compromisso entre: granularidade suficiente para distinguir tons, dimensionalidade moderada ($8 \times 3 = 24$ atributos), robustez a ruídos e variação de iluminação.

A normalização: $h = h/(h.sum() + 1e - 8)$ garante que variações de exposição, iluminação ou brilho não distorçam os histogramas.

2.1.3 Descritor RGB

Depois de calcular o histograma de cada um dos três canais, concatenamos cada canal em somente um, assim tendo 48 características. Na prática, esse descritor permite que modelos simples (como KNN ou Decision Tree) já capturem boa parte das diferenças visuais entre classes, enquanto modelos mais complexos (SVM, MLP e Random Forest) exploram melhor a riqueza estrutural do vetor.

2.1.4 Relação entre as Características e os Resultados Obtidos

Observando os resultados das matrizes de confusão (arquivos enviados), é possível verificar que:

- Classes com padrões de cor muito distintos (ex.: Marge e Maggie) tendem a se beneficiar do histograma RGB.
- Personagens visualmente mais próximos (ex.: Homer × Bart ou Lisa × Maggie) apresentaram confusões previsíveis, mitigadas parcialmente pelo ensemble.

Assim, o uso combinado de HOG + histogramas de cor foi fundamental para permitir que os classificadores produzissem resultados consistentes mesmo em uma base pequena e com baixa resolução.

2.2 Classificadores Avaliados

Para a etapa de classificação das características extraídas (e Histogramas de Cor), foram selecionados cinco algoritmos distintos, visando explorar diferentes comportamentos estatísticos. Os modelos escolhidos foram:

- **K-Nearest Neighbors (KNN)**: Um classificador baseado em instância que rotula amostras com base na classe majoritária entre os k vizinhos mais próximos.
- **Support Vector Machine (SVM)**: Focado em encontrar o hiperplano ótimo de separação em espaços de alta dimensão.
- **Decision Tree (DT)**: Um modelo baseado em regras de inferência hierárquicas, capaz de capturar relações não lineares simples.
- **Random Forest (RF)**: Um método de bagging que combina múltiplas árvores de decisão para reduzir a variância e o risco de overfitting.
- **Multilayer Perceptron (MLP)**: Uma rede neural artificial feedforward que utiliza retropropagação para aprender mapeamentos complexos entre as características visuais e as classes dos personagens.

2.2.1 Análise Comparativa dos Classificadores

Os experimentos demonstraram que a tarefa de classificação dos personagens apresenta desafios significativos, principalmente devido à natureza das características extraídas e ao possível desequilíbrio entre as classes.

Observa-se que, embora o SVM e o Random Forest tenham alcançado acurácia razoável (acima de 47%), seus valores de F1-Score Macro foram baixos (0,395 e 0,266, respectivamente). Isso indica um viés dos modelos em favor das classes majoritárias. A análise das matrizes de confusão confirma esse comportamento:

- O Random Forest classificou corretamente 33 das 35 amostras da classe majoritária (Classe 0), mas falhou completamente em identificar qualquer amostra das Classes 2, 3 e 4 (Recall = 0 para estas classes).
- O SVM apresentou comportamento similar, com alta precisão global, mas baixa capacidade de recuperação (Recall) para as classes minoritárias.

Por outro lado, o MLP e o KNN demonstraram um equilíbrio maior entre precisão e revocação, sacrificando um pouco da acurácia global para manter a capacidade de detectar as classes menos representadas, resultando em F1-Scores superiores aos métodos baseados em árvores.

Além dos classificadores individuais, foi implementada uma estratégia de Ensemble Learning utilizando um Soft Voting Classifier. Este ensemble combina as probabilidades preditas por 20 estimadores base (variações dos algoritmos citados acima com diferentes hiperparâmetros) para computar a classe final, buscando maior robustez e generalização.

Modelo final combinado:

- 5 KNNs
- 5 SVMs
- 4 Random Forests
- 3 Decision Trees
- 3 MLPs

Em votação **soft**, utilizando probabilidades médias.

2.2.2 Otimização de Hiperparâmetros e Validação Cruzada

Para garantir que os resultados não fossem enviesados por uma única divisão de dados ou por configurações arbitrárias, adotou-se um protocolo rigoroso de validação.

O processo de treino e validação seguiu a técnica de Validação Cruzada Estratificada (Stratified K-Fold) com **k=10**. Esta técnica garante que a proporção de classes em cada dobra seja preservada, o que é crucial dado o possível desbalanceamento entre as classes dos personagens.

Para cada algoritmo individual, foi executada uma busca exaustiva de hiperparâmetros (Grid Search), otimizando a métrica **F1-Score Macro**. As grades de busca definidas foram:

Tabela 2 – Tabela de Hiperparâmetros

Modelo	Hiperparâmetros
Decision Tree	k=3, 5, 7, 9Pesos: ‘uniform’ e ‘distance’
Random Forest	C=1, 10, 100 Gamma: ‘scale’, 1^3 e 1^4 Kernel: ‘rbf’
KNN	45,26%
SVM (RBF)	53,68%
MLP	51,58%

3 RESULTADO

A seguir iremos apresentar na Tabela 3 os resultados da validação cruzada com $k = 10$ em cada dos modelos, utilizando a base de dados de treino.

Tabela 3 – Tabela Resultado Validação Cruzada

Modelo	Acurácia	Média F1-score
<i>Decision Tree</i>	0,5885	0,5474
<i>Random Forest</i>	0,7566	0,7600
KNN	0,6947	0,6774
SVM (RBF)	0,7522	0,7501
MLP	0,7345	0,7402
<i>Ensemble</i>	0,7345	0,7219

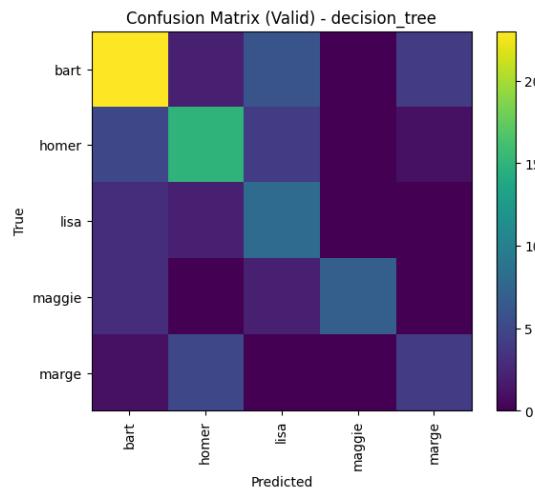
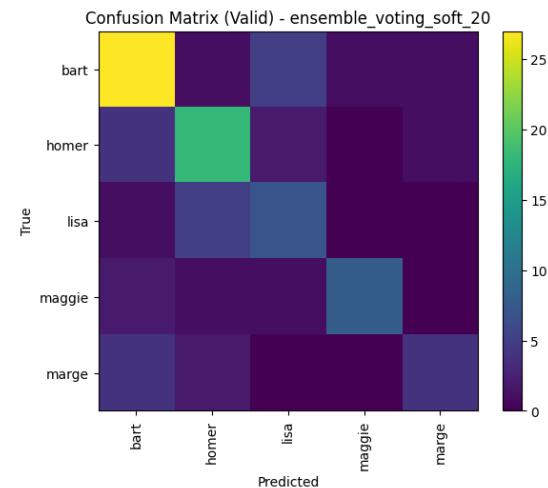
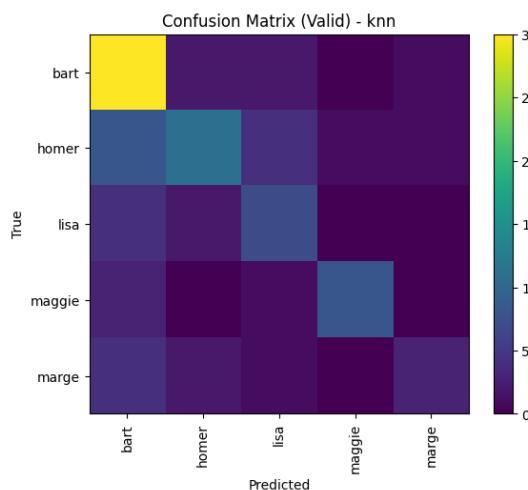
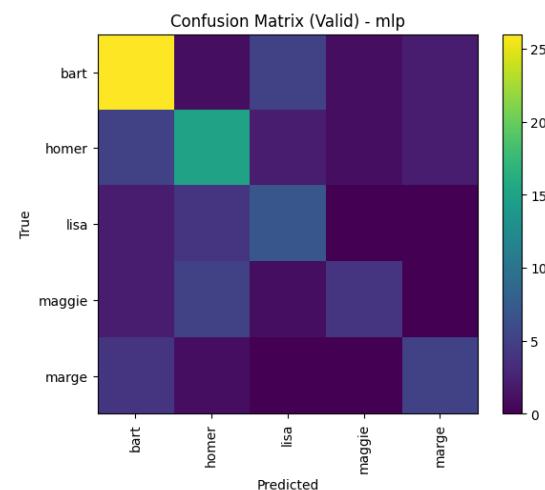
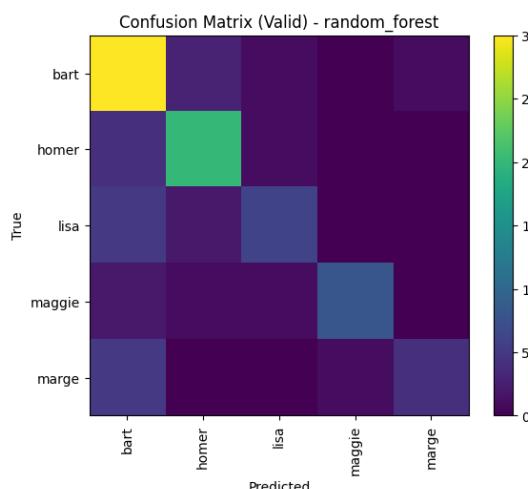
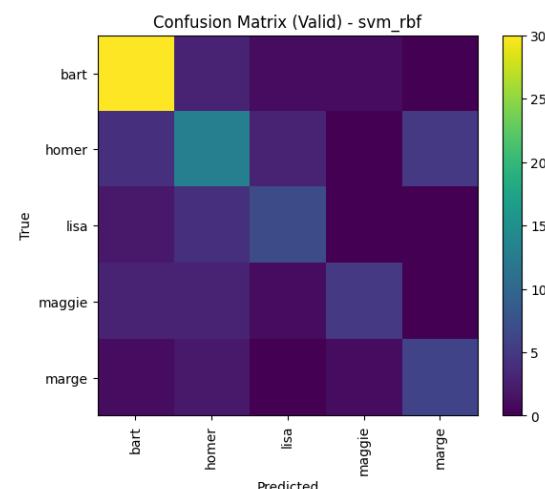
Ao analisarmos a tabela a cima, podemos observar que modelo que obteve o maior resultado foi o *Random Forest*, obtendo uma acurácia de 0,7566 e uma média de F1-score de 0,7600. A seguir na Tabela 4 podemos observar os resultados da classificação utilizando os modelos treinados com a base para treino, e testando com a base de teste.

Tabela 4 – Tabela Resultado Teste Final

Modelo	Acurácia	Média F1-score
<i>Decision Tree</i>	0,6000	0,5824
<i>Random Forest</i>	0,7158	0,6731
KNN	0,6211	0,5800
SVM (RBF)	0,6421	0,5955
MLP	0,6000	0,5523
<i>Ensemble</i>	0,6737	0,6388

Podemos observar na Tabela dos Resultados dos testes finais, que assim como nos testes de validação cruzada, o *Random Forest* conseguiu o melhor resultado tanto em acurácia quanto na média do F1-score.

A seguir temos as Figuras referentes a matriz de confusão de cada modelo conseguida com a avaliação do teste realizado.

Figura 1 – Matriz de Confusão Árvore de Decisão**Figura 2 – Matriz de Confusão do ensemble****Fonte: Autoria Própria.****Figura 3 – Matriz de Confusão KNN****Fonte: Autoria Própria.****Figura 4 – Matriz de Confusão MLP****Fonte: Autoria Própria.****Figura 5 – Matriz de Confusão Random Forest****Fonte: Autoria Própria.****Figura 6 – Matriz de Confusão SVM****Fonte: Autoria Própria.****Fonte: Autoria Própria.**

4 CONCLUSÃO

Este trabalho propôs resolver o seguinte problema proposto pelo professor: Desenvolver um código que, utilizando uma base de dados dos Simpsons, separada em um conjunto de teste e um de treino, extrair desta base de dados características, para utilizarmos estas características para treinar modelos de Inteligência Artificial (IA), dentro deles um método de combinação estática de classificadores.

Ao realizarmos o experimento, conseguimos que a melhor característica encontrada foi o RGB e que o modelo que conseguiu o melhor resultado foi o *Random Forest*.

4.1 Trabalhos Futuros

Sugerimos que para obter resultados melhores, utilizem alguma técnica de segmentação de imagens que produza características numéricas, assim utilizando não somente as cores como tambem os "contornos"para classificarmos os personagens.