

Documentação Main.py

Objetivo

Este script realiza o reconhecimento de números manuscritos a partir de uma imagem, utilizando um modelo treinado com a base de dados MNIST e a biblioteca Keras.

Importações

- `keras.models.load_model`: Carrega o modelo .h5 treinado.
- `PIL.Image`, `ImageOps`: Manipulação e processamento da imagem.
- `numpy`: Manipulação de arrays numéricos.
- `matplotlib.pyplot`: Visualização da imagem com o número reconhecido.

Função `prepare_image(path)`: prepara a imagem para o formato necessário ao modelo MNIST.

Parâmetros

- `path (str)`: Caminho para o arquivo de imagem.

Etapas de processamento

1. Abre a imagem e converte para escala de cinza.
2. Inverte as cores (dígito branco em fundo preto).
3. Redimensiona a imagem para 28x28 pixels.
4. Normaliza os valores de pixel para o intervalo [0, 1].
5. Adiciona dimensões: canal (28, 28, 1) e batch (1, 28, 28, 1)

Retorno

- `img_arr`: imagem pronta para o modelo.
- `img`: imagem invertida usada na visualização.

Carregamento do Modelo

```
model = load_model('mnist_number_recognition_keras.h5')
```

Preparo da Imagem

```
image_path = 'numero_teste.png'  
img_input, img_processed = prepare_image(image_path)
```

Predição

```
prediction = model.predict(img_input)  
predicted_number = np.argmax(prediction)
```

Exibição do Resultado

```
print(f"O número reconhecido é: {predicted_number}")
```

Visualização

A imagem é exibida com o número reconhecido como título da figura.

Requisitos

- Python 3.x
- Bibliotecas: keras, pillow, numpy, matplotlib Instalação: `pip install keras pillow numpy matplotlib`

Observações

- O modelo .h5 e a imagem devem estar no mesmo diretório.
- A imagem deve conter um único dígito manuscrito, centralizado e com boa visibilidade