

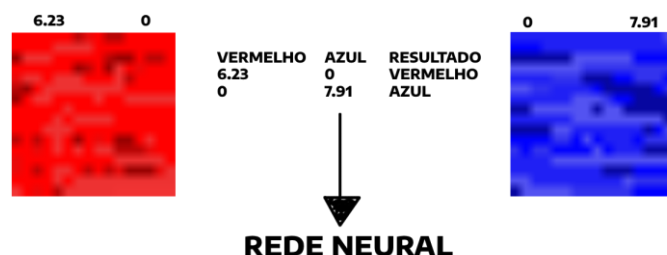
Redes neurais convolucionais

Usado para a visão computacional, como por exemplo carros autônomos na detecção de proximidade, tanto para ver carros, pedestres e outros.

Essas redes são utilizadas para processar imagens de maneira eficiente, extraindo padrões complexos a partir dos pixels presentes, sendo assim uma ótima ferramenta para o reconhecimento e classificação de imagens.

Sua função

Imagine duas imagens formadas por 16 pixels de altura e comprimento, formado com 256 pixels ao total, cada pixel é formado por uma cor RGB (uma mistura de vermelho, verde e azul). É extraído padrões da imagem que se diferenciam entre si, como por exemplo a quantidade de vermelho e azul.



Dificuldades

Ao utilizar uma rede neural comum para o reconhecimento dos pixels, rapidamente se torna inviável ao considerar imagens com tamanhos maiores, uma imagem 32x32 considerando 3 possíveis entradas de cor por pixel já possui 3072 possíveis entradas na qual acaba deixando o tempo de execução extenso.

Solução

A rede neural convolucional possui uma função para diminuir as entradas da rede neural (única área que ela atua de forma distinta), reconhecendo padrões e separando os pixels que possuem algum significado e aqueles que não possuem nenhuma função. A imagem passa por quatro etapas para filtrar essas entradas, sendo elas:

- Etapa 1: Operador de convolução

- Etapa 2: Pooling
- Etapa 3: Flattening
- Etapa 4: Rede neural densa

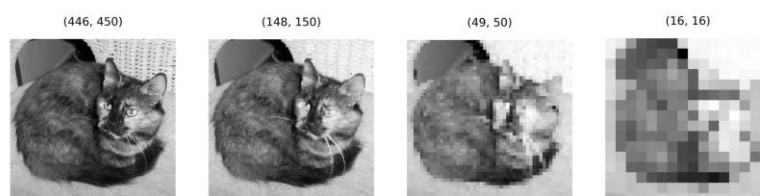
Operador de convolução

Consideramos a imagem original como uma matriz e multiplicamos uma outra matriz Kernel para obter diferentes "efeitos". A função mais comum é para a detecção de bordas.



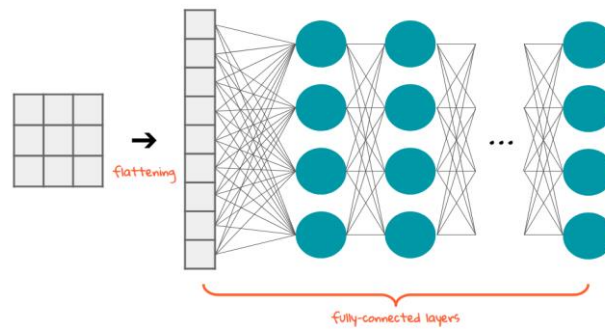
Pooling

É responsável por selecionar as características mais relevantes (reduzindo o overfitting e ruídos desnecessários), diferente do operador de convolução, o pooling retorna o maior valor de cada matriz que está lendo a fim de reduzir bastante a imagem.



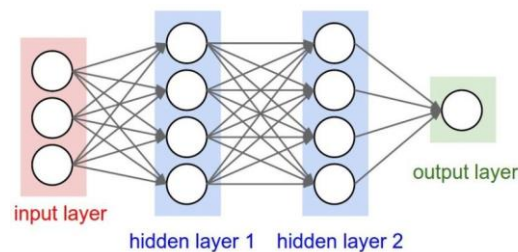
Flattening

O último pré processamento da imagem, partindo da matriz gerada pelo pooling, é reorganizado para um vetor, sendo cada valor deste vetor uma das entradas a ser utilizada na rede neural.



Rede neural densa

Com funcionamento semelhante as redes neurais já estudadas, a rede neural densa é responsável por separar o resultado, decidindo como exemplo se o valor escrito é 1, 3 ou 9. cada módulo da camada oculta é responsável por tomar essa decisão, a resposta será uma porcentagem de chance para cada resposta.



Atividade

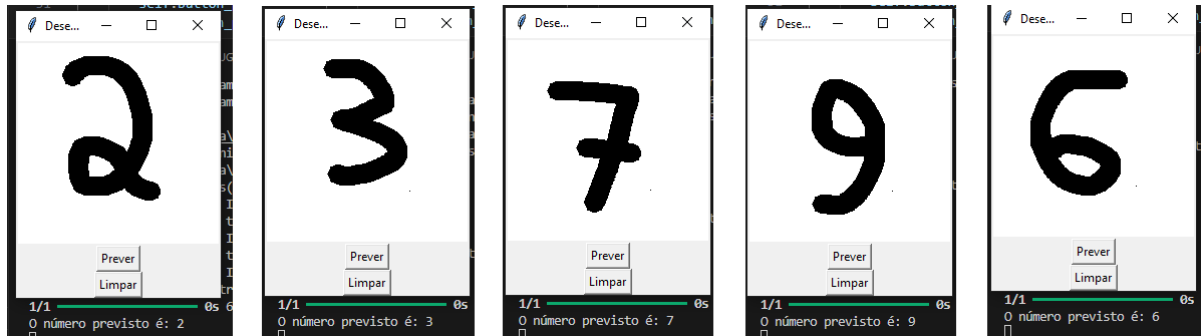
Na atividade da seção seguinte, utilizamos uma rede neural convolucional para identificar e classificar dígitos manuscritos. O exercício envolveu várias etapas, desde o pré-processamento das imagens, que foram normalizadas para garantir uniformidade, até a construção e treinamento do modelo com múltiplas camadas convolucionais, pooling e densas.

Atividade criada

Considerando o tema do card centralizado em adivinhar os números, a atividade criada parte do mesmo princípio, porém adicionando uma função nova para desenhar um número, utilizando da biblioteca tkinter para criar uma interface gráfica.

O código abre essa interface gráfica e transforma o valor desenhado para o formato de 28x28 para ser feito a predição, retornando no console o resultado que o modelo prever.

Utilizando de outro conceito anteriormente aprendido, para facilitar a execução do código, foi criada uma pasta separada para o treinamento do modelo e no código principal é apenas importado a rede neural. Ao remover a necessidade de treinar o modelo a cada execução, foi possível treinar o modelo usando um número elevado de épocas, melhorando no geral a sua eficiência e acurácia.



Conclusão

As redes neurais convolucionais são ferramentas poderosas para a visão computacional, usadas para tarefas como reconhecimento de imagens em carros autônomos. Elas processam imagens de forma eficiente, extraindo padrões complexos e reduzindo a dimensionalidade dos dados através de etapas como convolução, pooling, flattening e redes neurais densas. Essas técnicas superam as limitações das redes neurais tradicionais, que se tornam inviáveis para imagens maiores. No exemplo prático, uma rede neural convolucional foi utilizada para reconhecer dígitos manuscritos, com um modelo treinado previamente e uma interface gráfica para desenhar e prever números.