UNIVERSIDADE LUSÓFONA DE HUMANIDADES E TECNOLOGIA
CUL LISBOA

Thiago Gustavo Vieira de Paiva

RELATÓRIO - SEGMENTAÇÃO DE IMAGEM

Lisboa
2023

THIAGO GUSTAVO VIEIRA DE PAIVA

RELATÓRIO - SEGMENTAÇÃO DE IMAGEM

Relatório apresentado ao curso de Mestrado em Ciência de Dados da Universidade Lusófona de Humanidades e tecnologia, como parte necessária de avaliação da disciplina de Tópicos em Aprendizagem Automatizada e suas Aplicações.

Lisboa, 15 de Junho de 2023

Resumo

O presente relatório tem por objetivo detalhar as atividades realizadas no Projeto de Segmentação de Imagens que fazem uso de redes neurais convolucionais.

Os procedimentos adotados quanto a tarefas de preparação dos dados do *dataset* PASCAL VOC, seleção do modelo *DeepLab*, sua implementação com uso do framework PyTorch, treinamento do modelo e avaliação segundo métricas apropriadas ao problema e, por fim, exibição da segmentação realizada tomando como parâmetro a máscara de *ground-truth*.

Palavras-chave: segmentação de imagens, redes neurais convolucionais, PASCAL VOC, DeepLab, PyTorch, treinamento, avaliação.

Abstract

The present report aims to detail the activities carried out in the Image Segmentation Project that utilizes convolutional neural networks. The procedures adopted include tasks such as dataset preparation for PASCAL VOC, selection of the DeepLab model, its implementation using the PyTorch framework, model training, evaluation based on appropriate metrics for the problem, and finally, displaying the segmentation results compared to the ground-truth mask.

Keywords: image segmentation, convolutional neural networks, PASCAL VOC, DeepLab, PyTorch, training, evaluation.

Sumário

1 Introdução

A segmentação de imagem é uma tarefa fundamental no campo da visão computacional, que envolve a divisão de uma imagem em regiões semanticamente significativas. Essa técnica é amplamente utilizada em várias aplicações, como reconhecimento de objetos, detecção de bordas, segmentação de órgãos em imagens médicas, entre outros. A realização da segmentação precisa das imagens é um desafio importante, uma vez que é uma etapa crítica para a compreensão e interpretação automatizada das imagens.

Neste trabalho, a segmentação das imagens é feita com o uso do *dataset* Pattern Analysis Statistical Modelling and Computational Learning Visual Object Classes - PASCAL VOC [The PASCAL Visual Object Classes Challenge 2012].

Particularmente o problema de segmentação de imagens é tratado com o uso de Rede Neurais Convolucionais (CNNs). As CNNs têm se mostrado extremamente eficazes em tarefas de visão computacional, pois são capazes de aprender características relevantes diretamente das imagens, levando a resultados mais precisos e robustos.

O objetivo deste projeto é explorar a aplicação das CNNs para a segmentação de imagens utilizando o modelo DeepLab [Chen et al. 2017]. Serão realizadas etapas de preparação do dataset PASCAL VOC, desenvolvimento do modelo DeepLab ResNet-50 utilizando um framework de deep learning, treinamento do modelo e avaliação de sua performance. Além disso, serão utilizadas métricas relevantes, como Intersection over Union (IoU), F1 score e acurácia, para quantificar o desempenho da segmentação.

Ao final do projeto, espera-se obter um modelo de segmentação de imagens, capaz de segmentar corretamente objetos nas imagens do dataset.

2 Objetivos

2.1 Gerais

A construção de um modelo capaz de realizar a segmentação de imagens é o objetivo principal deste projeto.

2.2 Específicos

Como parte do processo de desenvolvimento deste projeto, é possível destacar como obetivos específicos:

1. Preparar os dados para o treinamento do modelo;
2. Treinamento do modelo DeepLab.
3. Avaliar quanto às métricas IoU, acurácia e F1-score.
4. Analisar predições.

3 Metodologia

A metodologia utilizada neste projeto de segmentação de imagens partiu da extração, carregamento e preparação dos dados. Seguiu então com a seleção do modelo DeepLab, instanciação e treinamento. Avaliação do modelo treinado para uma quantidade de épocas pequena. Treinamento do modelo para quantidade maior de épocas a fim de melhorar a segmentação das imagens e reavaliação quanto às mesmas métricas.

3.1 Extração, Carregamento e Preparação dos Dados

Inicialmente os dados do dataset PASCAL VOC [The PASCAL Visual Object Classes Challenge 2012] que possui 20 classes com dados para treino e validação com 11530 imagens contendo 27450 objetos anotados e 6929 segmentações.

Desta forma, primeiramente foi necessário realizar o carregamento dos dados. Em razão das dimensões serem diversas, uma transformação que permitisse redimensionar para a resolução de 256x256 pixels em todas as imagens foi aplicada bem como a transformação destas em tensores em conjuntos de treino e teste num ratio de 80/20 dos dados.

Tensor é uma estrutura de dados multidimensional semelhante a uma matriz ou dicionário python, que pode ser usada para representar dados sendo assim fundamental para armazenar e manipular dados no *framework* do PyTorch.

Os tensores suportam operações matemáticas e de álgebra linear além de funções de ativação e outras operações comuns em redes neuronais fora a capacidade de serem serializados para execução em *hardware* dedicado a exemplo de placas gr[aficas (GPUs) que, em virtude da possibilidade de acelera;'ao, podem facilitar imensamente a computação nestas.

Após realizadas as devidas transformações o desenvolvimento seguiu com a seleção e instanciação do modelo DeepLab.

3.2 Arquitetura de Rede Neural Convolucional DeepLab

O DeepLab [Chen et al. 2017] é uma arquitetura de rede neural convolucional (CNN) desenvolvida para a tarefa de segmentação semântica de imagens proposta por Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy e Alan L. Yuille.

O objetivo deste modelo é combinar uma rede neural convolucional com um campo receptivo atrous e um fator de escala global para obter segmentações refinadas. O campo

receptivo atrous permite que a rede neural capture informações contextuais em várias escalas, incorporando diferentes taxas de dilatação para aumentar o campo receptivo sem aumentar excessivamente os parâmetros da rede o que permite consequentemente discernir entre diferentes classes e delimitar as fronteiras dos objetos na imagem.

O modelo utilizado neste trablho foi uma variante conhecida por DeepLab ResNet-50 que se utiliza da arquitetura ResNet-50.

A arquitetura ResNet-50 [He et al. 2016] é uma rede neural convolucional profunda que foi proposta por Kaiming He, Xiangyu Zhang, Shaoqing Ren e Jian Sun.

A capacidade de treinar redes profundas com sucesso superando o problema de degradação de desempenho à medida que as redes ficam mais profundas permitiu à combinação destas arquiteturas a alcançar relevância em desafios de segmentação de imagem.

3.3 Treinamento

A fase de treinamento se deu com o conjunto de treino após as transformações o que foi possível com o uso do DataLoader do Pytorch. Apartir de então o modelo DeepLab ResNet-50 foi instanciado e colocado para o modo de treinamento.

A função de perda foi definida como a CrossEntropyLoss(), que é bastante usada em problemas de classificação. Neste caso, a máscara segmentada é tratada como o *label* de destino.

O otimizador *Adaptive Moment Estimation (Adam)* foi usado. Este otimizador mantém uma taxa de aprendizado adaptativa para cada parâmetro do modelo com base nas estimativas do gradiente acumulado e da média móvel exponencial dos gradientes ao quadrado (RMS).

O modelo foi em seguida treinado para um número específico de épocas igual a 10. Dentro de cada época, um loop é executado iterando sobre os lotes de treinamento do DataLoader.

O tamanho do lote (*batch size*)no *DataLoader* foi ajustado de acordo com a memória e cores de CPU disponíveis para o treinamento dada a impossibilidade durante a implmentação de se utilizar de uma placa de vídeo dedicada para o treino.

Cada lote teve o otimizador zerado (optimizer.zero_grad()) para evitar acúmulo de gradientes. As saídas do modelo foram obtidas passando as imagens de entrada onde a perda foi calculada ao comparar as saídas com as máscaras segmentadas. O gradiente foi calculado (loss.backward()) e os parâmetros do modelo foram atualizados atualizados (optimizer.step()).

Após o treinamento, o estado do modelo foi salvo usando a função torch.save() e iniciou-se a fase de avaliação para este treinamento.

Importante ressaltar que, após a avaliação do modelo para 10 épocas, o modelo foi submetido novamente a treinamento para 40 épocas e uma nova fase de avaliação.

3.4 Avaliação do modelo

Inicialmente, verificou-se as perdas do modelo treinado para 10 epochs, o que resultou em:

Época	**Loss**
1	0.24918922781944275
2	0.20213866421108037
3	0.19253508082993043
4	0.19026672070632217
5	0.18773083441733010
6	0.18433189445017464
7	0.18598272394938548
8	0.18361554193040713
9	0.18338388794921134
10	0.18133230568444142

Tabela 1 – Resultados de Loss por Época

A avaliação do modelo considerou as métricas de *Intersection over Union* (IoU), *accuracy* e *recall*.

A métrica IoU, também conhecida por Jaccard Index, é usada para avaliar a sobreposição entre duas áreas ou conjuntos e, no contexto de segmentação de imagens, mede a precisão da segmentação comparando a máscara segmentada predita com a máscara de referência (ground truth).

O cálculo envolve a área de sobreposição e a área de união entre as duas máscaras, sendo a fórmula dada por:

$$\text{IoU} = \frac{\text{Área de sobreposição}}{\text{Área de união}} \tag{3.1}$$

Fórmula: Índice de Jaccard (IoU).

A área de sobreposição é a interseção entre a máscara segmentada predita e a máscara de referência. Ela representa a região em que ambas as máscaras concordam.

A área de união é a região total coberta pelas duas máscaras, incluindo as áreas em que apenas uma das máscaras está presente.

O valor do IoU varia de 0 a 1, sendo 0 indicativo de nenhuma sobreposição entre as máscaras e 1 indicativo de uma correspondência perfeita. Um valor maior de IoU indica uma maior precisão da segmentação.

O IoU é especialmente útil em tarefas de segmentação de imagem, pois considera tanto a precisão quanto a completude da máscara segmentada. Ele fornece uma medida de quão bem a máscara segmentada se alinha com a máscara de referência.

Na prática, o IoU é calculado para cada objeto ou classe individualmente e, em seguida, uma média dos valores de IoU é tomada para obter uma medida geral do desempenho da segmentação.

O IoU é amplamente utilizado em tarefas de segmentação de imagem, como detecção de objetos, segmentação semântica e segmentação de instâncias. É uma métrica comumente relatada em trabalhos científicos e competições relacionadas a essas áreas, fornecendo uma avaliação quantitativa da qualidade da segmentação.

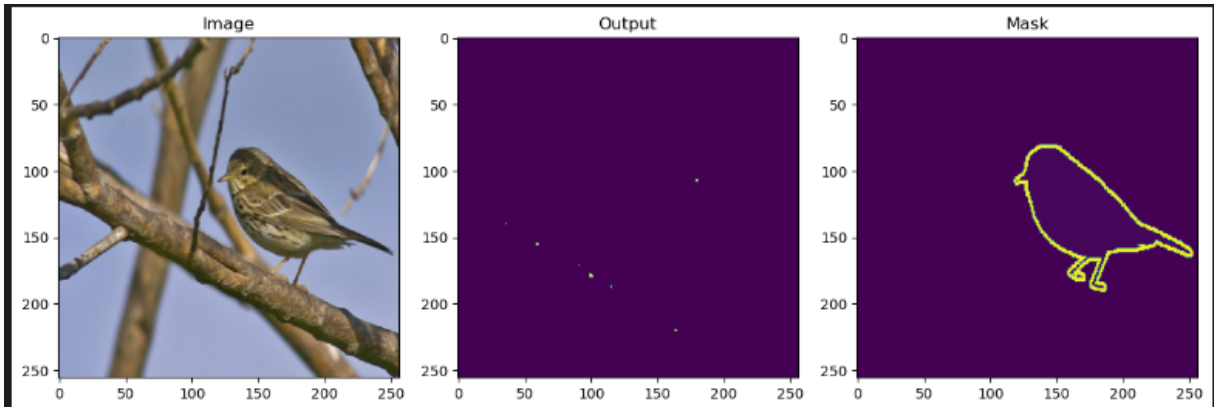
O valor médio de IoU, accuracy e recall encontrados para 10 épocas foi:

Métrica	**Valor Médio**
IoU	0.0008590079705382914
Acurácia	0.6985275407270952
Recall	112.1390621029247

Tabela 2 – Resultados das Métricas para 10 Épocas

Segue abaixo exemplo de entrada, saída e máscara usada como *ground truth*:

Figura 1 – Predição 10 épocas: *Input - Output - Mask*



Percebe-se que, apesar de o número de épocas pequeno não permite ainda predizer com muita eficácia a máscara.

O treinamento foi realizado novamentes para 40 épocas, totalizando em 50 a quantidade de vezes que o mesmo modelo foi submetido em treinamento e agora, para os valores de perda médio após o loop de treino, foi obtido:

Average Loss 50
0.07571765291196468

Tabela 3 – Resultados de Average Loss para 50 Épocas

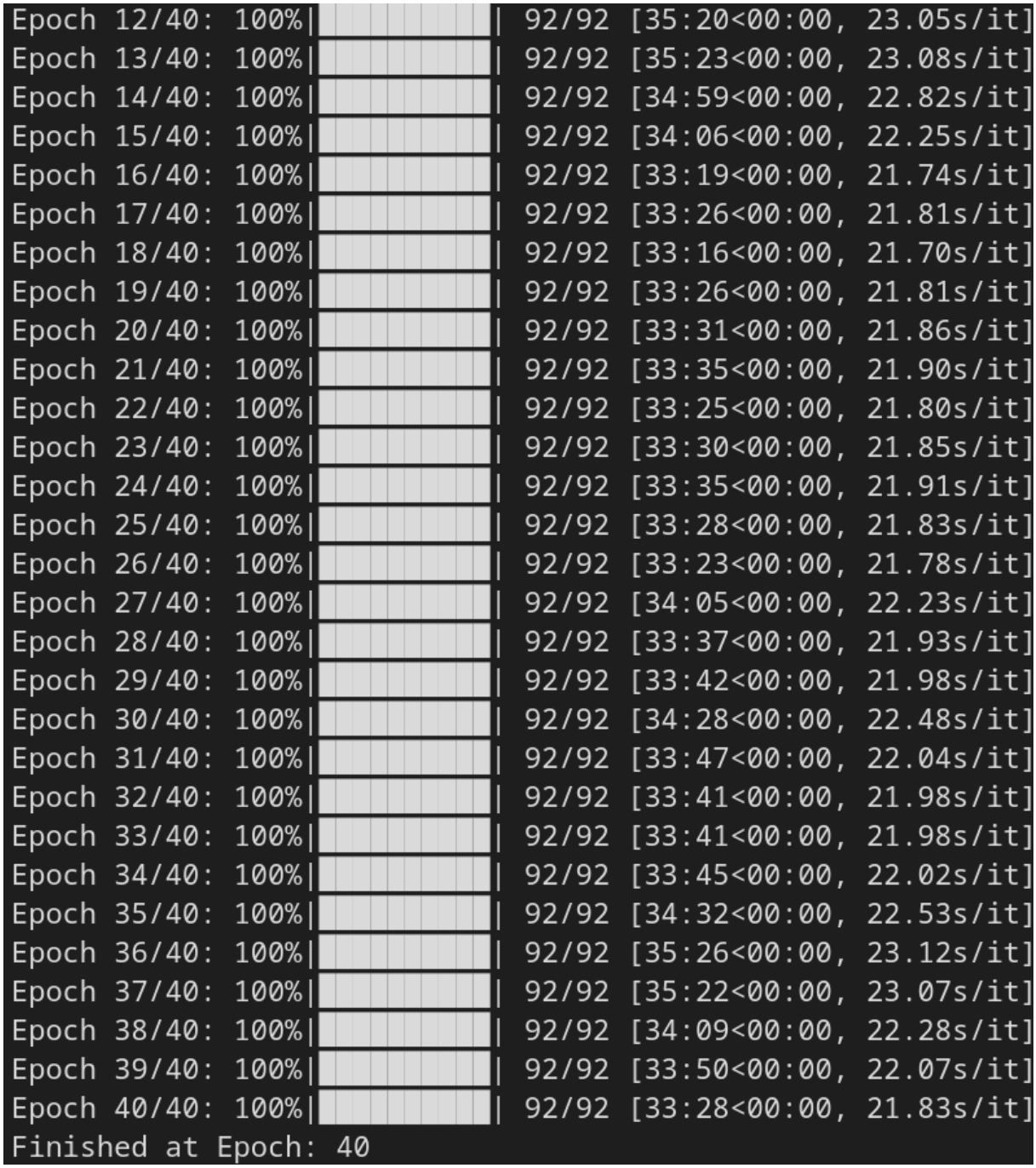
O que permitiu concluir que o modelo estava aprendendo e se ajustando melhor aos dados.

Importante aqui destacar que, apesar de o treinamento ter sido realizado com sucesso e o modelo também salvo inclusive com cópia de *backup*, no momento de avaliação deste, obteve-se erro apontando para a ausência de chaves no estado salvo de modo que a continuação da avaliação ficou prejudicada.

O tempo que o treinamento levou para 40 épocas pode em parte ser visualizado pela imagem a seguir:

O custo de treinamento longo sem recursos de hardware devidos impossibilitou a exibição do que de certa forma pôde ser verificado apenas com a avaliação sobre as perdas e esperava-se constatar também conforme as métricas anteriores.

Figura 2 – Tempo Treino de 40 épocas:



4 Considerações Finais

A proposta principal deste trabalho envolvia a segmentação de imagens a fim de predizer máscaras fazendo uso do modelo DeepLab ResNet-50 e avalar o treinamento ao longo de várias épocas para enfim avaliar quanto a métricas pertinentes a problemas de segmentação de imagem.

Entretanto, convém relatar a dificuldade encontrada em razão de não dispôr de hardware dedicado à esta tarefa de modo que o treinamento ao longo de 40 épocas teve uma média de duração de 36 minutos por época realizada apenas em cima da memória e CPU o que levou à demora na obtenção de resultados. Erros cometidos em quaisquer fase levaram a uma enorme perda em razão deste motivo.

Durante o desenvolvimento, as técnicas, competências e metodologia científica necessária foram desenvolvidas de modo que foi possível entender o funcionamento de redes neurais convolucionais (CNNs) o que leva à conclusão de que tanto o objetivo geral quanto os específicos foram atingidos.

Referências

CHEN, L.-C. et al. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. In: . [s.n.], 2017. Disponível em: <https://arxiv.org/abs/1606.00915>. Citado 2 vezes nas páginas 6 e 8.

HE, K. et al. Deep residual learning for image recognition. In: IEEE. *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.], 2016. p. 770–778. Citado na página 9.

The PASCAL Visual Object Classes Challenge. *The PASCAL VOC Dataset*. 2012. <http://host.robots.ox.ac.uk/pascal/VOC/>. Citado 2 vezes nas páginas 6 e 8.