Universidade do Estado do Rio de Janeiro

Centro de Tecnologia e Ciências

Instituto de Matemática e Estatística

Michel Pedro Filippo

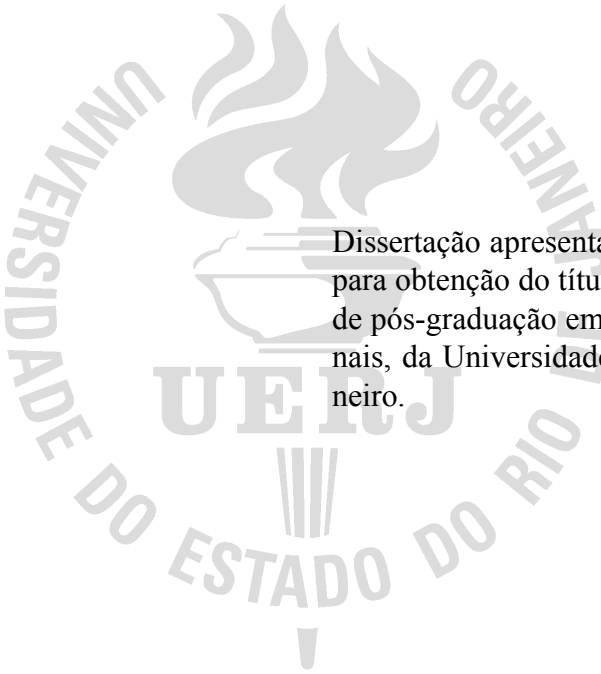
Aprendizagem profunda aplicada à caracterização de minérios: discriminando minerais opacos e não opacos de resina epóxi em imagens de microscopia ótica de luz refletida

Rio de Janeiro

2020

Michel Pedro Filippo

Aprendizagem profunda aplicada à caracterização de minérios: discriminando minerais opacos e não opacos de resina epóxi em imagens de microscopia ótica de luz refletida



Dissertação apresentada como requisito parcial para obtenção do título de Mestre, ao programa de pós-graduação em Ciências da Computacionais, da Universidade do Estado do Rio de Janeiro.

Orientadores: Prof. Dr. Gilson Alexandre Ostwald Pedro da Costa
Prof. Dr. Otávio da Fonseca Martins Gomes

Rio de Janeiro

2020

CATALOGAÇÃO NA FONTE
UERJ / REDE SIRIUS / BIBLIOTECA CTC-A

F483 Filippo, Michel P.

Aprendizagem profunda aplicada à caracterização de minérios: discriminando minerais opacos e não opacos de resina epóxi em imagens de microscopia ótica de luz refletida / Michel Pedro Filippo. - 2020.

68 f. :il.

Orientadores: Gilson Alexandre Ostwald Pedro da Costa e Otávio da Fonseca Martins Gomes.

Dissertação em Ciências da Computação - Universidade do Estado do Rio de Janeiro. Instituto de Matemática e Estatística.

1. Processamento de imagens - Teses. 2. Análise de imagem - Teses. 3. Microscopia ótica - Teses. 4. Minérios - Teses. I. Costa, Gilson Alexandre Ostwald Pedro da II. Gomes, Otávio da Fonseca Martins III. Universidade do Estado do Rio de Janeiro. Instituto de Matemática e Estatística. IV. Título.

CDU 004.932

Patricia Bello Meijinhos CRB7/5217 - Bibliotecária responsável pela elaboração da ficha catalográfica

Autorizo, apenas para fins acadêmicos e científicos, a reprodução total ou parcial desta dissertação.

_____ _____

Assinatura Data

Michel Pedro Filippo

Aprendizagem profunda aplicada à caracterização de minérios: discriminando minerais opacos e não opacos de resina epóxi em imagens de microscopia ótica de luz refletida

Dissertação apresentada como requisito parcial para obtenção do título de Mestre, ao programa de pós-graduação em Ciências da Computacionais, da Universidade do Estado do Rio de Janeiro.

Aprovada em 09 de dezembro de 2020.

Banca Examinadora:

_____

Prof. Dr. Gilson Alexandre Ostwald Pedro da Costa (Orientador)
Instituto de Matemática e Estatística - UERJ

_____

Prof. Dr. Otávio da Fonseca Martins Gomes (Orientador)
Centro de Tecnologia Mineral e PPGeo/Museu Nacional/UFRJ - CETEM

_____

Prof. Dr. Guilherme Lucio Abelha Mota
Instituto de Matemática e Estatística - UERJ

_____

Prof. Dr. Sidnei Paciornik
Pontifícia Universidade Católica do Rio de Janeiro

_____

Prof. Dr. Marcos Vinicius Colaço Gonçalves
Instituto de Física - UERJ

Rio de Janeiro

2020

RESUMO

FILIPPO, Michel Pedro. *Aprendizagem profunda aplicada à caracterização de minérios:* discriminando minerais opacos e não opacos de resina epóxi em imagens de microscopia ótica de luz refletida. 2020. 68 f. Dissertação (Mestrado em Ciências da Computação) – Instituto de Matemática e Estatística, Universidade do Estado do Rio de Janeiro, Rio de Janeiro, 2020.

A discriminação entre minerais não opacos e a resina de embutimento em imagens de microscopia de luz refletida de amostras de minério é um problema desafiador e bem documentado. A refletância especular semelhante desses materiais dificulta sua discriminação, mesmo por especialistas humanos. Embora leves diferenças visuais, como reflexões internas e diferenças sutis na superfície polida, possam ajudar os humanos a delinear partículas distintas desses materiais, as técnicas convencionais de processamento de imagem não são capazes de capturar tais características (*features*) subjetivas e tendem a falhar, tornando-se um problema carente de uma solução computacional robusta. Inspirado no recente sucesso de técnicas de aprendizagem profunda (*deep learning*) na interpretação de imagens, o presente trabalho avalia a eficácia da segmentação semântica de imagens de minérios por meio de modelos de aprendizagem profunda, na discriminação entre a resina epóxi de embutimento e partículas de minério contendo minerais opacos e não opacos. Neste trabalho é avaliado o desempenho da arquitetura DeepLabv3+ e algumas variantes são propostas a fim de melhorar a precisão da segmentação, particularmente nas fronteiras das partículas minerais. Os modelos de aprendizagem profunda foram avaliados usando quatro conjuntos de dados distintos, contendo imagens de diferentes minérios, adquiridos com diferentes configurações experimentais. Os resultados mostraram desempenhos excelentes, sistematicamente acima de 90% de *Overall Accuracy* e *F1 Score*, e até 94% para alguns conjuntos de dados. Além disso, a fim de analisar a capacidade de generalização da solução de aprendizagem profunda, avaliações de validação cruzada foram conduzidas, usando um dos quatro conjuntos de dados para treinar o modelo e testando-o nos outros conjuntos de dados. Possivelmente, este trabalho apresenta a primeira abordagem de segmentação semântica baseada em aprendizagem profunda para a discriminação de minerais opacos e não opacos de resina epóxi em imagens de microscopia de luz refletida.

Palavras-chave: Aprendizagem profunda. Análise de imagem. Segmentação semântica. Microscopia de minério. Minério de ferro.

ABSTRACT

FILIPPO, Michel Pedro. *Deep learning applied to ore characterization:* discriminating opaque and non-opaque epoxy resin minerals in reflected light optical microscopy images. 2020. 68 f. Dissertação (Mestrado em Ciências da Computação) – Instituto de Matemática e Estatística, Universidade do Estado do Rio de Janeiro, Rio de Janeiro, 2020.

Discrimination between non-opaque minerals and embedding resin in reflected light microscopy images from ore samples is a challenging and well-documented problem. The similar specular reflectance of these materials makes it difficult to discriminate, even by human experts. Although slight visual differences, such as internal reflections and subtle differences in the polished surface, can help humans to delineate distinct particles of these materials, conventional image processing techniques are unable to capture such subjective features and tend to fail, becoming a problem for which a robust computational solution was still missing. Inspired by the recent success of deep learning techniques in image interpretation, the present work evaluates the effectiveness of semantic segmentation of ore images through deep learning models, in the discrimination between embedding epoxy resin and ore particles containing opaque and non-opaque minerals. In this work, the performance of the DeepLabv3+ architecture is evaluated and some variants are proposed in order to improve segmentation accuracy, particularly at the boundaries of mineral particles. The deep learning models were evaluated using four different data sets, containing images of different ores, acquired with different experimental configurations. The results showed outstanding performances, systematically over 90% Overall Accuracy and F1 Scores, and up to 94% for some datasets. Additionally, in order to analyze the generalization capacity of the deep learning solution, cross-validation evaluations were conducted, using one out of the four datasets for training the model, and testing it on the other datasets. Possibly, this work presents the first approach of semantic segmentation based on deep learning for the discrimination of opaque and non-opaque epoxy resin minerals in reflected light microscopy images.

Keywords: Deep learning. Image analysis. Semantic segmentation. Ore microscopy. Iron ore.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ASPP	Atrous Spatial Pyramid Pooling
BSE	Elétrons retroespalhados (do inglês: Backscattered Electrons)
CLEM	Luz Correlativa e Microscopia Eletrônica (do inglês: Correlative Light and Electron Microscopy)
CNN	Convolutional Neural Network
DA	Domain Adaptation
DL	Deep Learning
EDS	Energy Dispersive X-ray Spectrometer
FCN	Fully Convolutional Network
FN	False Negative
FP	False Positive
MEV	Microscopia Eletrônica de Varredura
OA	Overall Accuracy
OS	Output Stride
PCA	Principal Components Analysis
SGD	Stochastic Gradient Descent
TN	True Negative
TP	True Positive

SUMÁRIO

INTRODUÇÃO

O constante crescimento da capacidade computacional disponível tem possibilitado o uso de técnicas cada vez mais sofisticadas, mas custosas computacionalmente, para a resolução de problemas complexos. Uma delas é o *Deep Learning* (DL), uma forma de aprendizado de máquina que permite aos computadores aprender com a experiência e entender problemas em termos de uma hierarquia de conceitos. Nos últimos anos, o DL têm se popularizado imensamente graças ao seu alto desempenho preditivo, muitas vezes superior ao de técnicas tradicionais de aprendizado de máquina. Além disso, hardwares capazes de processar algoritmos de DL estão cada vez mais acessíveis, contribuindo ainda mais para a popularização dessas técnicas, notadamente na área acadêmica.

Esses algoritmos procuram explorar a estrutura desconhecida na distribuição da entrada com o objetivo de descobrir boas representações de características (*features*) de alto nível definidas em termos de características de baixo nível. O objetivo é tornar essas representações de alto nível mais ricas semanticamente, mas robustas a variações locais (não informativas), enquanto preservam coletivamente o máximo possível das informações da entrada (BENGIO, 2012). Isso permite que um sistema aprenda funções complexas mapeando, diretamente dos dados, a entrada para a saída, sem depender de representações idealizadas por especialistas (*hand-craft*). Desta forma, um algoritmo de DL permite que uma rede neural aprenda hierarquias de informações de uma forma semelhante à função do cérebro humano (HEATON, 2020). De modo geral, o DL criou arquiteturas de redes neurais que abrangem mais camadas ocultas do que seus antecessores costumavam ter. Como consequência de seu design, o DL pode modelar problemas muito complexos.

Técnicas de DL vem sendo cada vez mais utilizadas em diversos segmentos. Na medicina, por exemplo, a utilização dessas técnicas vem crescendo constantemente: (LITJENS et al., 2017) analisa os principais conceitos de DL pertinentes à análise de imagens médicas e resume mais de 300 contribuições para o campo. Já (SHEN; WU; SUK, 2017) discute os fundamentos de métodos de DL e revisa seus sucessos no registro de imagens, detecção de estruturas anatômicas e celulares, segmentação de tecidos, entre outros. Além disso, com os avanços dos modelos de DL, especialmente redes neurais convolucionais (CNNs), o desempenho da classificação de

imagens de sensoriamento remoto foi significativamente melhorado devido às poderosas representações de características aprendidas por meio dessas CNNs (CHENG et al., 2018).

Mais recentemente as redes neurais profundas vêm sendo usadas para detecção de *fake news* em redes sociais que são, hoje, uma das principais fontes de notícias para milhões de pessoas em todo o mundo devido ao seu baixo custo, fácil acesso e rápida divulgação. No entanto, isso tem o custo de confiabilidade duvidosa e risco significativo de exposição à 'notícias falsas', escritas intencionalmente para enganar os leitores. Para isso (MONTI et al., 2019) propôs um novo modelo de detecção automática de notícias falsas com base em *Geometric* DL. Os algoritmos são uma generalização das CNNs clássicas, permitindo a fusão de dados distintos, como conteúdo, perfil e atividade do usuário, gráfico social e propagação de notícias.

Avanços recentes em DL, especialmente CNNs profundas, levaram, também, à uma melhoria significativa das técnicas de segmentação semântica: que atribuem uma classe à cada pixel da imagem de entrada. Muitas áreas de aplicação em ascensão utilizam mecanismos de segmentação precisos e eficientes, como: direção autônoma, navegação interna, e realidade virtual ou aumentada. Essa demanda coincide com o surgimento de abordagens de DL em quase todos os campos relacionados à visão computacional (GARCIA-GARCIA et al., 2018).

As microscopias de luz transmitida e refletida, respectivamente para minerais transparentes e opacos, são provavelmente as técnicas mais tradicionais de identificação mineralógica. Durante os últimos dois séculos, diversos métodos analíticos baseados em várias propriedades dos minerais foram desenvolvidos e refinados. No entanto, esses métodos tradicionais geralmente requerem um mineralogista especialista e apenas alguns deles podem ser aplicados em sistemas automatizados. Assim, nas últimas décadas, o microscópio eletrônico de varredura (MEV) tornou-se a principal ferramenta usada para microscopia de minérios. Foram desenvolvidos sistemas de mineralogia automatizada que identificam minerais e realizam rotinas de quantificação por meio de um software de análise de imagens integrado a fim de determinar a assembleia mineralógica, quantificar as fases presentes e medir a liberação mineral (GOMES; PACIORNIK, 2012).

A possibilidade de usar uma solução de análise de imagem baseada apenas na microscopia de luz refletida é de grande interesse para a indústria mineral, devido à sua maior simplicidade

e seu menor custo de instalação e operação, especialmente em áreas de mina. Para atingir esse objetivo, o principal desafio é a segmentação de minerais não opacos da resina de embutimento em imagens de microscopia de luz refletida. Este trabalho enfrenta este desafio por meio de métodos de DL.

Especificamente no campo da análise de imagens, as arquiteturas de DL de destaque correspondem às CNNs, nomeadas assim por suas camadas convolucionais. Ao revisar a literatura, pode-se notar que, quando comparado à visão humana, o desempenho das CNNs supera o homem em várias aplicações de classificação de imagens (STALLKAMP et al., 2012). As arquiteturas das CNNs são versáteis e foram adotadas para diversas finalidades, entre as quais estão classificação de imagens, detecção de objetos e segmentação semântica. Consequentemente, a quantidade de trabalhos de pesquisa que aplicam DL na ciência dos materiais cresceu substancialmente nos últimos dois anos (e.g., (IGLESIAS; SANTOS; PACIORNIK, 2019); (SVENSSON, 2019); (DECOST et al., 2019); (AZIMI et al., 2018); (JIANG et al., 2018); (KONDO et al., 2017)).

Este trabalho é dedicado à segmentação semântica, baseada em DL, de imagens de microscopia de luz refletida, visando a discriminação de minerais opacos e não opacos da resina epóxi. Mais especificamente, é empregada uma arquitetura específica de DL *Fully Convolutional Network* (FCN), o DeepLabv3+ (CHEN et al., 2018), para a tarefa, e são propostas e avaliadas algumas variantes do modelo original do DeepLabv3+, que visam aprimorar o nível de detalhe, particularmente nas bordas das partículas minerais, e na melhoria da precisão discriminativa.

Os modelos de DL propostos são avaliados em quatro conjuntos de dados diferentes, contendo imagens de diferentes minérios, adquiridas com diferentes configurações experimentais. Além disso, para analisar a capacidade de generalização do modelo com melhor desempenho, considerando a classificação dos conjuntos de dados individuais, é realizada uma avaliação intragrupo, de validação cruzada do modelo, usando um dos quatro conjuntos de dados disponíveis para o treinamento do modelo. Finalmente, é realizada uma avaliação de validação cruzada entre grupos, na qual o modelo é treinado com amostras que pertencem a mais de um conjunto de dados e avaliado com amostras dos outros conjuntos de dados.

Este é, possivelmente, o primeiro trabalho que emprega segmentação semântica baseada em aprendizado profundo para o problema de discriminação de minerais opacos e não opacos da resina epóxi em imagens de microscopia de luz refletida.

Em resumo, as principais contribuições deste trabalho são:

- Avaliação de modelos de segmentação semântica totalmente convolucionais, baseados na arquitetura DeepLabv3+, na discriminação de minerais opacos e não opacos da resina epóxi em imagens de microscopia de luz refletida.
- Proposição de uma extensão da arquitetura do DeepLabv3+ para essa tarefa específica.
- Avaliação das variantes do DeepLabv3+ usando quatro conjuntos de dados com imagens de diferentes minérios, com diferentes configurações de aquisição.
- Análise da capacidade de generalização da abordagem proposta por meio de experimentos de validação cruzada considerando os diferentes conjuntos de dados.
- Utilização de imagens de MEV co-registradas que são espacialmente correlacionadas às imagens de microscopia de luz refletida para conceber um procedimento objetivo e reproduzível para geração de dados de referência.

O restante desta dissertação está organizada da seguinte forma: Na seção 1 são apresentados os trabalhos relacionados. A seção 2 apresenta alguns conceitos fundamentais para uma melhor compreensão dos métodos utilizados neste trabalho. Já a seção 3 descreve os métodos de DL utilizados para a classificação bem como a descrição das bases de dados utilizadas. A seção 4 é destinada à descrever os experimentos realizados, definir as métricas utilizadas para avaliar os modelos, descrever a preparação das bases de dados e detalhar a parametrização da rede. A seção 5 é dedicada aos resultados obtidos por cada um dos métodos utilizados no estudo e, além disso, são discutidos cada um dos resultados. Por fim, são apresentadas as conclusões a respeito dos resultados obtidos e proposições para trabalhos futuros.

1 REVISÃO BIBLIOGRÁFICA

Esta seção apresenta uma revisão bibliográfica sobre a utilização de *Deep Learning* (DL) na análise de imagens na indústria de mineração, ciência de materiais e engenharia. Nesse contexto, a maioria das pesquisas em DL são focadas na classificação e segmentação de imagens. Assim, o restante desta seção resume as abordagens de classificação de imagens e de segmentação semântica encontradas na literatura.

Em termos gerais, as abordagens atualmente disponíveis não diferem significativamente das aplicações em outros campos. Na verdade, duas estratégias principais são implementadas na ciência de materiais ao lidar com a classificação de imagens baseada em DL. Primeiro, o esquema convencional de DL ponta a ponta (*end-to-end*), que explora todas as camadas de rede, incluindo seu classificador embutido. A segunda corresponde à chamada caracterização por CNN. Essa estratégia deriva características (*features*) dos mapas de ativação (*feature maps*) de uma CNN, produzindo representações de imagens em uma ou várias escalas, na sequência um classificador separado discrimina objetos com base nessas características. Diferenças entre as abordagens acima mencionadas trazem consequências para o processo de treinamento. Assim, embora as CNNs sejam treinadas de ponta a ponta, a caracterização da CNN depende de um classificador treinado posteriormente. Além disso, como pode ser observado na literatura, o uso de estratégias de aprendizagem por transferência (*transfer learning*) é bastante usual em abordagens de caracterização.

Masci et al. (MASCI et al., 2012) apresentam as pesquisas mais antigas que utilizam DL aplicada à pesquisa em ciência dos materiais. Dedicada ao problema de reconhecimento de defeitos em imagens de inspeção de superfície de aço, a abordagem emprega uma arquitetura que consiste em uma sequência de blocos convolucionais e *max-pooling*, seguidos por uma camada totalmente conectada e, por fim, uma camada de saída *softmax*. Em uma pesquisa mais recente abordando o mesmo problema, (YI; LI; JIANG, 2017) propôs o uso de uma arquitetura CNN semelhante, mas mais profunda. Além disso, (ZHANG et al., 2019a) investigou um procedimento de inspeção dedicado a peças de manufatura aditiva. Uma arquitetura análoga foi aplicada em (IGLESIAS; SANTOS; PACIORNIK, 2019) para classificar recortes (*patches*) de imagens de microscopia ótica de minérios de ferro, discriminando entre recortes de quartzo e recortes de

resina.

A caracterização baseada em CNN de imagens MEV para análise microestrutural foi investigada em (LING et al., 2017). Mapas de ativação de um codificador VGG-16 foram usados como atributos que são classificados por um classificador *random forest*. Três conjuntos de dados distintos, ligas de titânio, diversos tratamentos térmicos de aço e imagens de MEV sintéticos de materiais em pó, foram usados para validação. Outra abordagem de caracterização via CNN é apresentada em (ZHANG et al., 2019b), com foco na discriminação de classes de minerais em imagens microscópicas. A arquitetura DL consiste no modelo Inception-v3, que é, através de uma abordagem de aprendizagem por transferência, adaptado para a produção de atributos que são classificados com vários classificadores alternativos. Liu et al. (LIU et al., 2019) aplicou a caracterização baseada em CNN para identificação de minerais de rocha usando imagens fotográficas de curta distância. O método propõe um modelo abrangente combinando os atributos produzidos pelo Inception-v3 e um modelo de cores derivado usando o algoritmo de agrupamento *k-means*.

Kondo et al. (KONDO et al., 2017) exploraram CNNs para realizar regressão não linear de imagens MEV, com o objetivo de estimar a condutividade iônica de cerâmicas de zircônia estabilizadas com ítria. Devido às restrições de dados de treinamento, a arquitetura empregada consiste em uma versão simplificada da rede VGG. Além disso, CNNs foram explorados por (LI et al., 2020) para estimar a localização e alcalinidade das pelotas de minério de ferro através da análise de imagens de microscopia ótica. A abordagem é baseada no AlexNet como *backbone* e usa análise de componentes principais (PCA) para integrar atributos (*features*) rasos (*shallow*) ou profundas, produzindo um conjunto de características multiescala. As feições obtidas são submetidas à camadas totalmente conectadas, o que produz inferências de localização e alcalinidade.

Jiang et al. (JIANG et al., 2018) propõem um método para segmentação de grãos tomando imagens multi-ângulo de arenito como entrada. A abordagem começa agrupando imagens em superpixels usando propriedades espectrais e espaciais. Em seguida, uma CNN extrai atributos convolucionais multiescala dos superpixels. Finalmente, considerando tais características juntamente com outros atributos, um algoritmo de agrupamento difuso mescla superpixels vizinhos

em grãos.

As pesquisas sobre segmentação semântica de DL no campo da ciência dos materiais são bastante recentes. Abordando o estudo de rochas de arenito, (KARIMPOULI; TAHMASEBI, 2019) propuseram o uso da arquitetura SegNet para segmentação microtomográfica. Svensson (SVENSSON, 2019) comparou modelos distintos de CNNs para segmentação semântica de imagens de microscopia ótica de pelotas de minério de ferro. Para a discriminação das oito fases presentes nas imagens, a pesquisa empregou os modelos PSPNet, FC-DenseNet, DeepLabv3+, BiSeNet e GCN. Bezerra et al. (BEZERRA; AUGUSTO; PACIORNIK, 2020) propôs o uso de segmentação semântica em imagens de microtomografia de pelotas de minério de ferro usando o modelo de CNN U-Net. O estudo teve como objetivo discriminar entre poros, fissuras e classes de sólidos. Duan et al. (DUAN et al., 2019) empregou segmentação semântica para controle de qualidade na produção de pelotas verdes de minério de ferro. Os autores avaliaram uma versão leve da U-Net para a segmentação de imagens de pelotização em tons de cinza.

Uma abordagem de DL para segmentação semântica de minério de ferro bruto, aplicada a imagens de inspeção de correias transportadoras e pilhas de detonação é apresentada em (LIU et al., 2020). O método compreende três etapas: pré-processamento, segmentação e pós-processamento. No estágio de segmentação, as arquiteturas de segmentação semântica U-Net e ResUnet são empregadas para detecção e otimização de contorno. Lorenzoni et al. (LORENZONI et al., 2020) emprega uma U-Net para a análise de compósitos à base de cimento de endurecimento por deformação microestrutural em mircrotomografia computadorizada. Esta aplicação requer segmentação precisa das diferentes fases do material e outras características, o que representa uma tarefa complexa para algoritmos de segmentação convencionais. Evsevleev et al. (EVSEVLEEV; PACIORNIK; BRUNO, 2020) empregou a U-Net para segmentação semântica de compósitos de matriz de metal em tomografia computadorizada de raios-X síncrotron. A segmentação resultante é um insumo para a análise microestrutural desses compósitos, sendo a chave para o entendimento de seu comportamento micromecânico.

Azimi et al. (AZIMI et al., 2018) propôs um método de DL para classificação dos constituintes microestruturais em imagens de ligas de aço de baixo carbono. A arquitetura reproduz as camadas iniciais do codificador VGG-16. DeCost (DECOST et al., 2019) propôs uma abor-

dagem de DL de segmentação semântica para análise microestrutural de micrografias de aço carbono ultra-alto. A arquitetura usada é uma variante do PixelNet, cujo codificador segue a rede VGG-16, e o decodificador faz uma amostra dos mapas de ativação finais. Em sequência, os atributos associados a cada pixel são discriminados por meio de uma rede Multilayer Peceptron (MLP) de duas camadas.

Quando contrastado com as abordagens baseadas em DL aplicadas à ciência dos materiais encontradas na literatura, o trabalho apresentado nesta dissertação contém algumas diferenças importantes. Em primeiro lugar, enquanto esta abordagem discrimina entre minério (mais outros materiais de ganga) e resina no nível de pixel, a abordagem mais semelhante (IGLESIAS; SANTOS; PACIORNIK, 2019), que considera a discriminação entre quartzo e imagem de resina, produz um único resultado para recortes (*patches*) de imagem selecionados. Em segundo lugar, quase todas as abordagens de segmentação semântica mencionadas anteriormente usam referências produzidas por meio de inspeção visual, o que traz algum nível de subjetividade. Os dados de referência utilizados neste trabalho, no entanto, foram derivados de uma fonte de dados independente, imagens MEV espacialmente correlacionadas com os dados de microscopia ótica, através de um procedimento de microscopia correlativa objetiva e reprodutível. Adicionalmente, são propostas extensões de uma arquitetura original de DL com o objetivo de melhorar seu desempenho no contexto do problema-alvo, enquanto a maioria dos trabalhos relacionados se limita à exploração de arquiteturas padrão, nem mesmo tentando ajustar seus hiperparâmetros. Também são apresentados resultados quantitativos de várias rodadas de treinamento para todas as variantes de arquitetura, a fim de levar em consideração a natureza estocástica dos modelos de DL. Além disso, investigou-se a generalidade da abordagem proposta, avaliando seu desempenho em diferentes conjuntos de dados, associados a diferentes tipos de minérios e composições minerais. Também foram realizados experimentos de validação cruzada, treinando e testando a abordagem com imagens dos diferentes conjuntos de dados na tentativa de avaliar sua capacidade de generalização. Por fim, investigou-se a capacidade do modelo proposto, avaliando seu desempenho quando submetido à um treinamento com imagens de domínios diferentes simultaneamente.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Classificação e segmentação semântica de imagens

A segmentação e classificação de objetos fazem parte do processamento de imagem baseado em aprendizado de máquina para treinamento de algoritmos de IA por meio da visão computacional. A classificação de imagens refere-se à tarefa de atribuir um ou mais rótulos (classes) globais à uma determinada imagem. Desta forma, uma imagem pode ser classificada como contendo primordialmente algum objeto ou classe de objeto, i.e., gato, cachorro, quartzo, resina, etc.

Em contrapartida, a segmentação semântica de imagens é o processo de classificação de todos os pixels em uma imagem pertencente a uma determinada classe. O objetivo da segmentação semântica é atribuir um rótulo a cada pixel de uma imagem de forma que os pixels com o mesmo rótulo compartilhem certas características *features*. Por exemplo, se houver 2 gatos em uma imagem, a segmentação semântica dará o mesmo rótulo a todos os pixels de ambos os gatos. Ela é usada principalmente para localizar objetos e limites como linhas e curvas nas imagens. A segmentação semântica é útil para detectar e classificar objetos em uma imagem quando houver mais de uma classe presente.

Desse modo, pensando em obter um melhor desempenho computacional, algumas arquiteturas chamadas de *Fully Convolutional Networks* (FCN) (LONG; SHELHAMER; DARRELL, 2014) foram idealizadas. Essas arquiteturas, normalmente, usam o modelo *encoder-decoder*. No *encoder* (codificador) a informação de entrada é transformada por camadas convolucionais e de *pooling* (vide as próximas duas seções), que sequencialmente reduzem as dimensões espaciais dos dados de entrada até uma camada chamada de *bottleneck*. A partir daí, no *decoder* (decodificador), camadas de de-convoluções (mais propriamente chamadas de convoluções transpostas), ou interpolações bilineares, recuperam sequencialmente as dimensões espaciais originais da imagem.

2.2 **Convolução e Convolução dilatada**

A convolução é uma operação matemática entre duas funções. Ela é definida como a integral do produto desses funções, depois que uma é deslocada, fornecendo como resultado uma nova função que expressa como a forma de uma é modificada pela outra.

No contexto das Rede Neurais Convolucionais (CNNs), uma convolução é uma operação linear que envolve a multiplicação de um conjunto de pesos entre duas matrizes bidimensionais. A multiplicação é realizada entre uma matriz de entrada e uma matriz bidimensional de pesos, chamada de filtro ou *kernel*, normalmente uma matriz menor e com o mesmo número de bandas da matriz de entrada. Este *kernel* "desliza" sobre os dados de entrada, realizando um produto elementar e, em seguida, produzindo um único pixel de saída referente ao somatório desse produto.

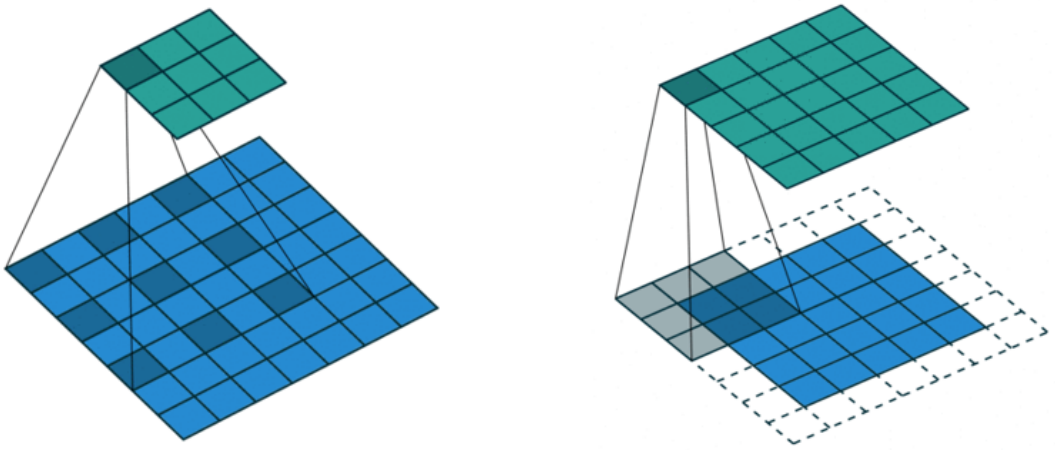
As CNNs utilizam uma série de operações convolucionais com o objetivo de extrair características, em vários níveis, da imagem de entrada e produzir um mapa probabilístico para cada pixel. O *kernel* repete este processo para cada local que "desliza", convertendo uma matriz bidimensional de características em outra matriz bidimensional de características. As características de saída são, essencialmente, as somas ponderadas (com os pesos sendo os valores do próprio kernel) das características da entrada localizadas aproximadamente no mesmo local do pixel de saída na camada de entrada. Isso significa que o tamanho do *kernel* determina diretamente quantas características da entrada serão combinadas na produção de um novo recurso da saída.

Para esse processo, que considera o contexto para a classificação de cada pixel da imagem, o ideal seria utilizar filtros grandes para a convolução, de forma a aumentar o seu campo receptivo e capturar um maior contexto. Entretanto, isso também aumentaria muito o número de operações realizadas pelas convoluções, aumentando muito o custo computacional e tornando o processo muitas vezes inviável.

A primeira versão do modelo Deeplab foi proposto por Chen et al. (CHEN et al., 2014), esse modelo introduziu uma implementação particular do *"hole algorithm"* (MALLAT, 1999), que foi concebido para o cálculo eficiente da transformada *wavelet* indecimada e tornou-se conhecido no campo de DL pelos termos convolução *atrous* ou dilatada. As convoluções dilatadas têm a capacidade de ampliar o campo de visão (receptivo) dos filtros convolucionais tradicio-

nais, incorporando assim contextos espaciais maiores sem aumentar o número de parâmetros ou a quantidade de cálculos. Isso é feito adicionando espaços vazios (zeros) entre os pesos desses filtros. É possível ver a diferença entre os filtros de convolução na Figura 1.

Figura 1 – Diferença entre a convolução dilatada (i.e., *Atrous Convolution*) e a convolução convencional

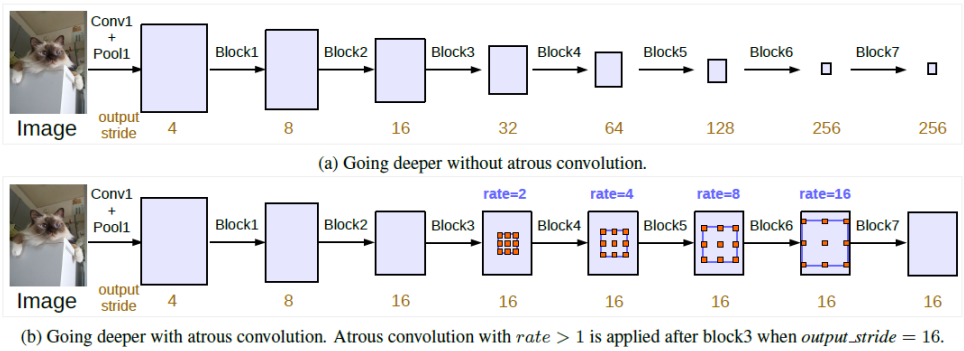


(a) *Atrous Convolution* (b) Convolução Convencional

Fonte: https://towardsdatascience.com/review-drn-dilated-residual-networks-image-classification-semantic-segmentation-d527e1a8fb5

O DeepLab utiliza uma arquitetura semelhante a de uma rede neural convolucional profunda (DCNN) (VGG-16 (SIMONYAN; ZISSERMAN, 2014) ou ResNet-101 (HE et al., 2015)), treinada para classificação de imagens e destinada à tarefa de segmentação semântica. Entretanto, todas as camadas totalmente conectadas são transformadas em camadas convolucionais, e as convoluções convencionais, a partir de um determinado ponto, são substituídas por camadas de convolução dilatada, o que aumenta a resolução a partir dessas camadas, no sentido em que as dimensões dos mapas de ativação subsequentes não diminuem da maneira em que aconteceria com convoluções convencionais. Isso é exemplificado na Figura 2.

Figura 2 – Diferença da arquitetura sem *Atrous Convolution* e com *Atrous Convolution*



(a) Going deeper without atrous convolution.

(b) Going deeper with atrous convolution. Atrous convolution with $rate > 1$ is applied after block3 when $output\_stride = 16$.

Fonte: (CHEN et al., 2017a)

2.3 **Camadas de Pooling**

As camadas convolucionais em uma rede neural convolucional aplicam os filtros às imagens de entrada para criar mapas de ativação (*feature maps*) que resumem a presença das suas características (*features*) na entrada. Ao empilhar camadas convolucionais em modelos profundos permitimos que camadas próximas à entrada aprendam características de baixo nível (por exemplo, linhas) e camadas mais profundas no modelo aprendam características de alto nível ou mais abstratas, como formas ou objetos específicos.

Uma limitação da saída do mapa de ativação de camadas convolucionais é que eles registram a posição exata das características na entrada. Isso significa que pequenos movimentos na posição das características na imagem de entrada resultarão em um mapa de ativação diferente. Isso pode acontecer com recorte, rotação, deslocamento e outras pequenas alterações na imagem de entrada.

Uma abordagem comum para resolver esse problema é chamada de amostragem reduzida (*down sampling*). Nela é criada uma resolução mais baixa de um sinal de entrada que ainda contém os elementos estruturais grandes ou importantes, sem os detalhes finos que podem não ser tão úteis para a tarefa.

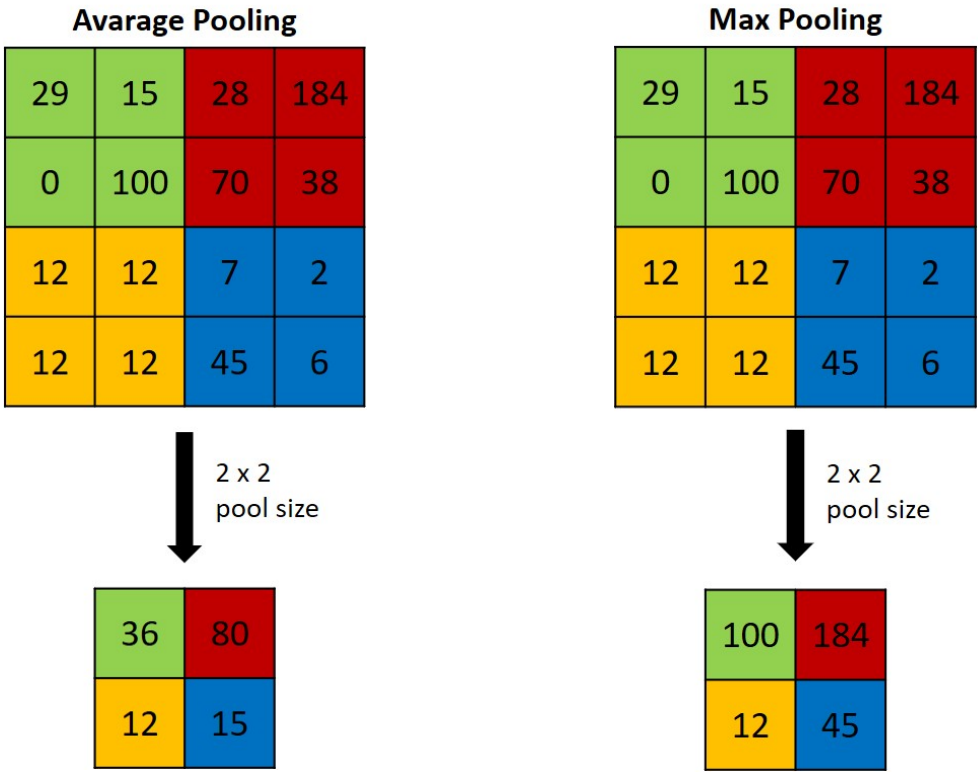
O *down sampling* pode ser obtido com camadas convolucionais, alterando-se o *stride* (passo) da convolução na imagem. Uma abordagem mais robusta e comum é usar uma camada de *pooling*. Uma camada de *pooling* é uma nova camada adicionada após a camada convolucional. Essa nova camada opera em cada mapa de ativação separadamente para criar um novo conjunto

com o mesmo número de mapas de ativação agrupadas. O tamanho da operação de pooling, assim como no *kernel* das camadas convolucionais, é menor que o tamanho do mapa de ativação. Duas funções comuns usadas nas operações de *pooling* são:

- *Pooling* médio: Calcula o valor médio para cada recorte (*patch*) no mapa de ativação.
- *Pooling* máximo: Calcula o valor máximo para cada recorte (*patch*) do mapa de ativação.

A figura 3 mostra a diferença das duas funções de *pooling* mais comuns.

Figura 3 – Diferença nas funções de *pooling* médio e máximo com tamanho 2



(a) *Pooling* médio (b) *Pooling* máximo

O resultado do uso de uma camada de *pooling* é uma versão resumida das características detectadas na entrada. Eles são úteis, pois pequenas mudanças na localização de características detectadas pela camada convolucional resultarão em um mapa de ativação agrupado com as características no mesmo local.

2.4 **Batch Normalization**

A ideia por trás do *batch normalization* é a mesma utilizada nas camadas de entrada. Por exemplo, quando há uma imagem de entrada na escala de 0 a 255, comumente essa imagem é normalizada para a escala de 0 a 1 de modo a acelerar o aprendizado e beneficiar a camada de entrada da rede. Tendo isso em vista, aplicar uma técnica semelhante para os valores das camadas ocultas, que estão mudando constantemente, beneficiaria a rede como um todo e consequentemente aumentaria, ainda mais, a velocidade de treinamento.

Desse modo, o *batch normalization* é proposto como uma técnica para ajudar a coordenar a atualização de várias camadas no modelo. Ele faz isso dimensionando a saída da camada, ou seja, padronizando as ativações de cada variável de entrada por *mini-batch*.

Padronizar as ativações da camada anterior significa que as suposições que a camada subsequente faz sobre a propagação e distribuição das entradas durante a atualização de peso não mudarão, ou mudarão muito pouco. Isso tem o efeito de estabilizar e acelerar o processo de treinamento das redes neurais profundas.

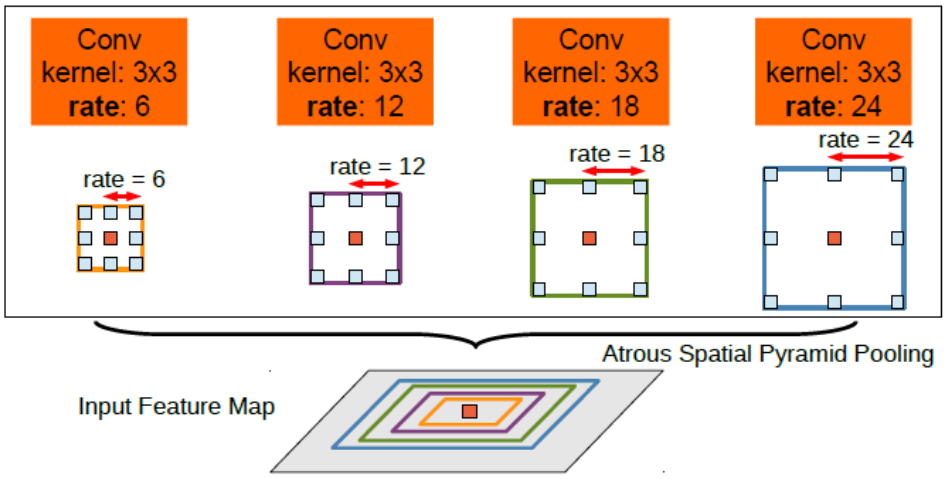
A normalização das entradas para a camada tem efeito no treinamento do modelo, reduzindo drasticamente o número de épocas necessárias. Ela também pode ter um efeito de regularização, reduzindo o erro de generalização.

Para aumentar a estabilidade de uma rede neural, o *batch normalization* normaliza a saída de uma camada de ativação anterior subtraindo a média do *batch* e dividindo pelo seu desvio padrão.

2.5 **Atrous Spatial Pyramid Pooling (ASPP)**

Uma segunda versão do modelo DeepLab foi proposta em (CHEN et al., 2017b). Sua grande novidade foi o chamado *Atrous Spatial Pyramid Pooling* (ASPP), que concatena o resultado de uma sequência de convoluções dilatadas com diferentes taxas de dilatação. As características (*features*) geradas por cada convolução são processadas paralelamente e depois fundidas para gerar o resultado final. A ideia da ASPP é exibida na Figura 4.

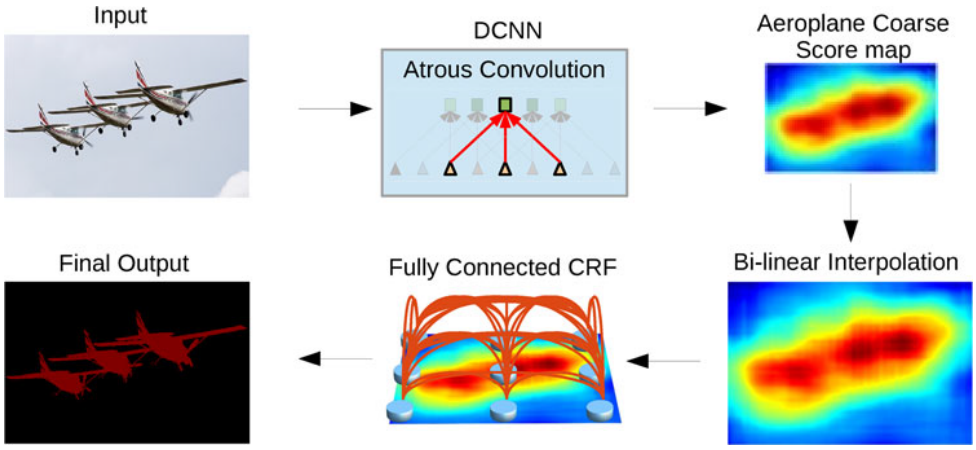
Figura 4 – *Atrous Spatial Pyramid Pooling* (ASPP)



Fonte: (CHEN et al., 2017b)

As duas primeiras versões do modelo DeepLab também continham um *Conditional Random Field* (CRF) totalmente conectado (SUTTON; MCCALLUM, 2006) anexado à saída da rede convolucional, que pretendia aprimorar o nível de detalhe do resultado final dos modelos. As etapas de processamento do CRF pode ser vista na Figura 5. O componente CRF foi descartado no terceiro módulo DeepLab (CHEN et al., 2017a), e o componente ASPP foi aumentado nessa versão usando características de nível de imagem produzidos por *image pooling* (LIU; RABINOVICH; BERG, 2015), que codifica o contexto global da imagem.

Figura 5 – Cadeia de processamento da rede

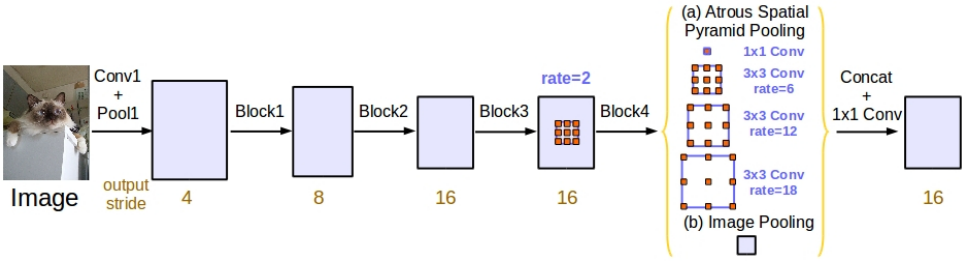


Fonte: (CHEN et al., 2017b)

O ASPP com diferentes taxas de dilatação captura efetivamente informações em várias escalas. No entanto, foi descoberto que, à medida que a taxa de amostragem (*sampling rate*)

aumenta, o número de pesos válidos do filtro se torna menor. No caso extremo em que o valor da taxa está próximo ao tamanho do mapa de ativação de entrada, um filtro 3 x 3, em vez de capturar o contexto da imagem, degenera para um filtro 1 x 1 simples, pois apenas o peso central do filtro é efetivo. Para superar esse problema, a terceira versão do DeepLab (CHEN et al., 2017a), incorpora informações de contexto global ao modelo semelhantes à ParseNet (LIU; RABINOVICH; BERG, 2015) e à PSPNet (ZHAO et al., 2016). Especificamente, é aplicado um *average global pooling* no último mapa de ativação do modelo, seguido de uma convolução 1 x 1 com 256 filtros (e *batch normalization*) e, posteriormente, é realizado um *upsample* através de uma interpolação bilinear para a dimensão espacial desejada. No final, o novo ASPP aprimorado consiste em uma convolução 1 x 1 e três convoluções 3 x 3 com taxas iguais a 6, 12 e 18 (todas com 256 filtros e *batch normalization*). As características resultantes são concatenadas e passam por outra convolução 1 x 1 (também com 256 filtros e *batch normalization*) antes da convolução final, 1 x 1. Resumindo, a diferença entre a segunda e terceira versão é a inclusão de *batch normalization* (IOFFE; SZEGEDY, 2015) no ASPP, para facilitar o treinamento, e a extração de características globais. A arquitetura do DeepLab v3 é exibida na Figura 6.

Figura 6 – Arquitetura do DeepLab V3



Fonte: (CHEN et al., 2017a)

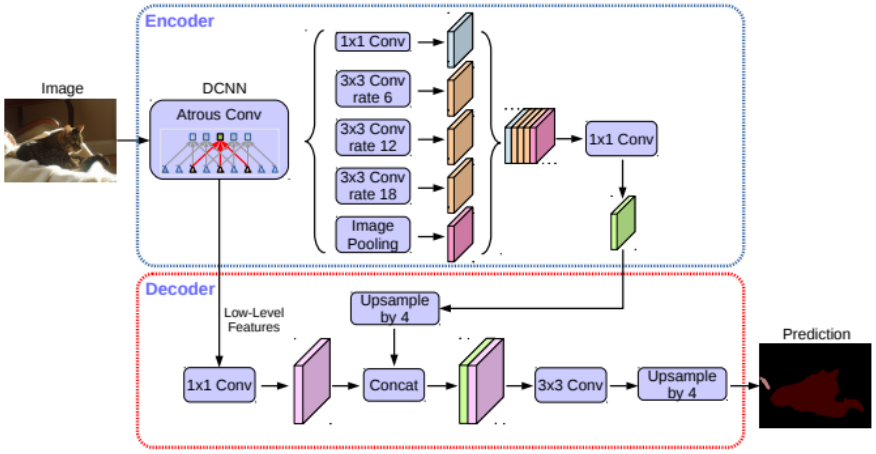
2.6 **Encoder-decoder**

Finalmente, o modelo DeepLabv3+ (CHEN et al., 2018) adota uma estrutura do tipo *encoder-decoder*. O *encoder* (codificador) segue o modelo DeepLabv3 e seu resultado é o último mapa de ativação. Um módulo de *decoder* (decodificador) simples foi desenvolvido para melhorar os resultados da segmentação, especialmente ao longo das bordas do objeto.

O resultado do *encoder* é geralmente calculado com um *output stride* de 16. O *output stride* é a razão entre a resolução espacial da imagem de entrada e a resolução espacial do mapa de

features de saída. A fim de recuperar detalhes de segmentação do objeto que não são recuperados por meio de uma interpolação bilinear simples por um fator de 16, o mapa de ativação de saída do *encoder* em DeepLabv3+ é primeiro interpolado bilinearmente por um fator de 4 e, em seguida, concatenado com o mapa de ativação de baixo nível correspondente do *backbone* da rede que possui a mesma resolução espacial. Antes da concatenação, uma convolução 1 x 1 é aplicada às características (*features*) de baixo nível para reduzir o número de canais associados a essas características, de modo que eles não superem a importância da saída do *encoder*. Após a concatenação, são aplicadas convoluções 3 x 3, finalmente seguidas por outra interpolação bilinear simples por um fator de 4 (CHEN et al., 2018).A Figura 7 mostra a arquitetura mais recente do DeepLab.

Figura 7 – Arquitetura do DeepLab V3+



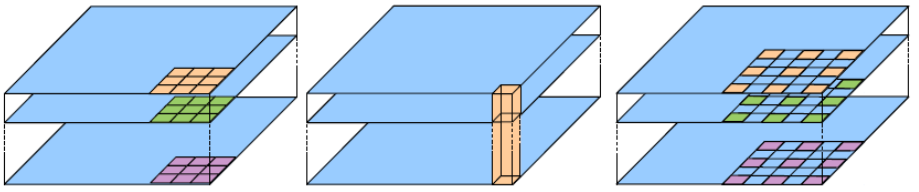
Fonte: (CHEN et al., 2018)

2.7 **Atrous Separable Convolution**

Em (CHEN et al., 2018) os autores também utilizaram o modelo Xception (CHOLLET, 2017) com as modificações propostas em (DAI et al., 2017) como *backbone* da rede de codificadores. As operações de *max pooling* no modelo Xception original também foram substituídas por *Atrous Separable Convolution* com *stride*, e *batch normalization* (IOFFE; SZEGEDY, 2015) e função de ativação ReLU foram adicionadas após cada de convolução em profundidade 3 x 3, como no projeto MobileNet (HOWARD et al., 2017). A *Atrous Separable Convolution* consiste em fatorar uma convolução padrão em convoluções menores (*depthwise convolution*)

seguida por uma convolução em ponto (ou seja, convolução 1 x 1), o que reduz drasticamente a complexidade da computação. Especificamente, a convolução em profundidade executa uma convolução espacial independentemente para cada canal de entrada (CHEN et al., 2018). A Figura 8 exemplifica a técnica empregada na rede.

Figura 8 – *Atrous Separable Convolution*



Fonte: (CHEN et al., 2018)

2.8 **Skip Connection**

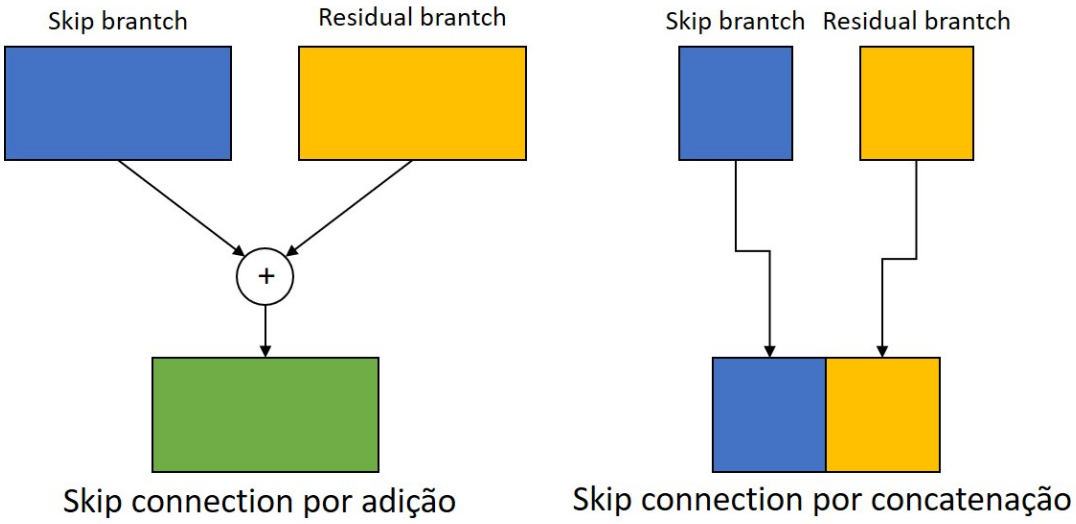
Atualmente, *skip connection* (conexão de salto) é um módulo padrão em muitas arquiteturas convolucionais. Usando uma *skip connection*, fornecemos um caminho alternativo para o gradiente (com *backpropagation*). É validado experimentalmente que esses caminhos adicionais são frequentemente benéficos para a convergência do modelo. *Skip connections* em arquiteturas profundas, como o próprio nome sugere, pulam algumas camadas na rede neural e alimentam a saída de uma camada como entrada para outras camadas (em vez de apenas a subsequente).

Em geral, existem duas maneiras fundamentais de usar as *skip connections* por meio de diferentes camadas não sequenciais:

- Adição como em arquiteturas residuais
- Concatenação como em arquiteturas densamente conectadas

A figura 9 mostra essas duas maneiras de usar as *skip connections*.

Figura 9 – Diferença das *Skip connection* por adição e concatenação



Neste trabalho a variante do modelo proposta foi criada usando *skip connections* com concatenação, seguindo os padrões da *skip connection* já existente na arquitetura do DeepLabv3+. Desse modo, as informações de baixo nível podem ser compartilhadas entre a entrada e a saída da rede.

3 MÉTODO

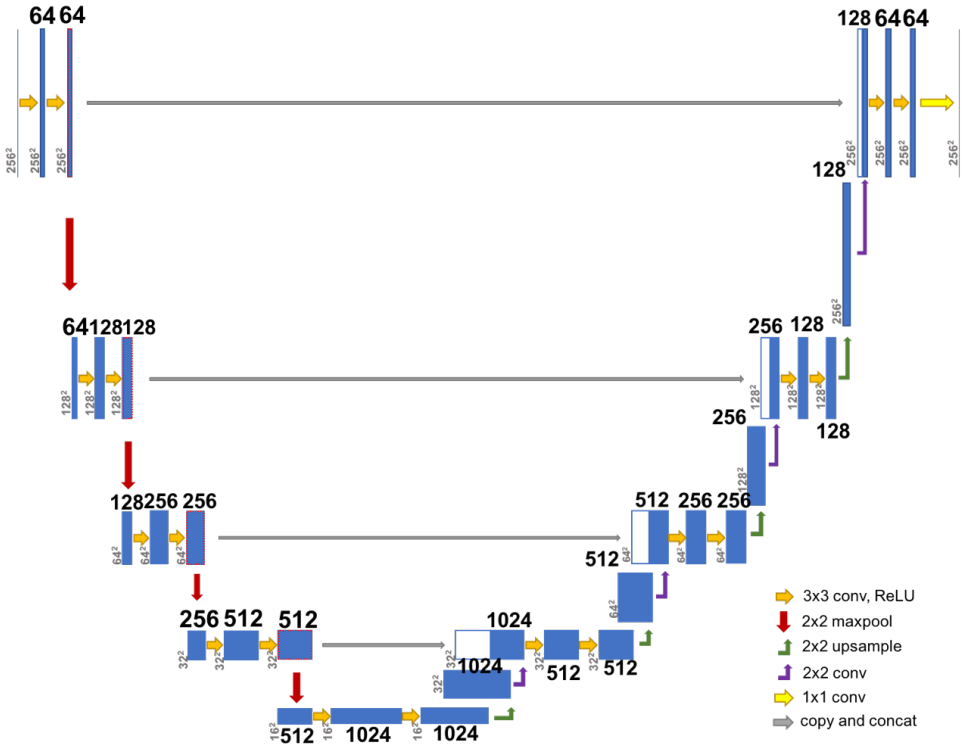
Esta seção apresenta as arquiteturas avaliadas neste trabalho, bem como suas diferenças. Além disso serão descritas todas as bases de dados utilizadas e as bibliotecas que deram suporte para a realização dos experimentos.

3.1 Arquitetura do DeepLabV3+ e alterações propostas

Os modelos mais atuais de aprendizagem profunda totalmente convolucionais para segmentação semântica seguem uma arquitetura do tipo *encoder-decoder*. O *encoder* realiza uma redução da dimensão espacial da entrada por meio de uma sequência de operações de convolução, geralmente, mas não necessariamente, seguida por operações de *pooling*. A parte do *decoder* da arquitetura, conectada à última camada do codificador (geralmente chamada de *bottleneck*), executa uma série de operações de *upsampling*, que podem ser implementadas por meio de convoluções transpostas e/ou interpolações bilineares.

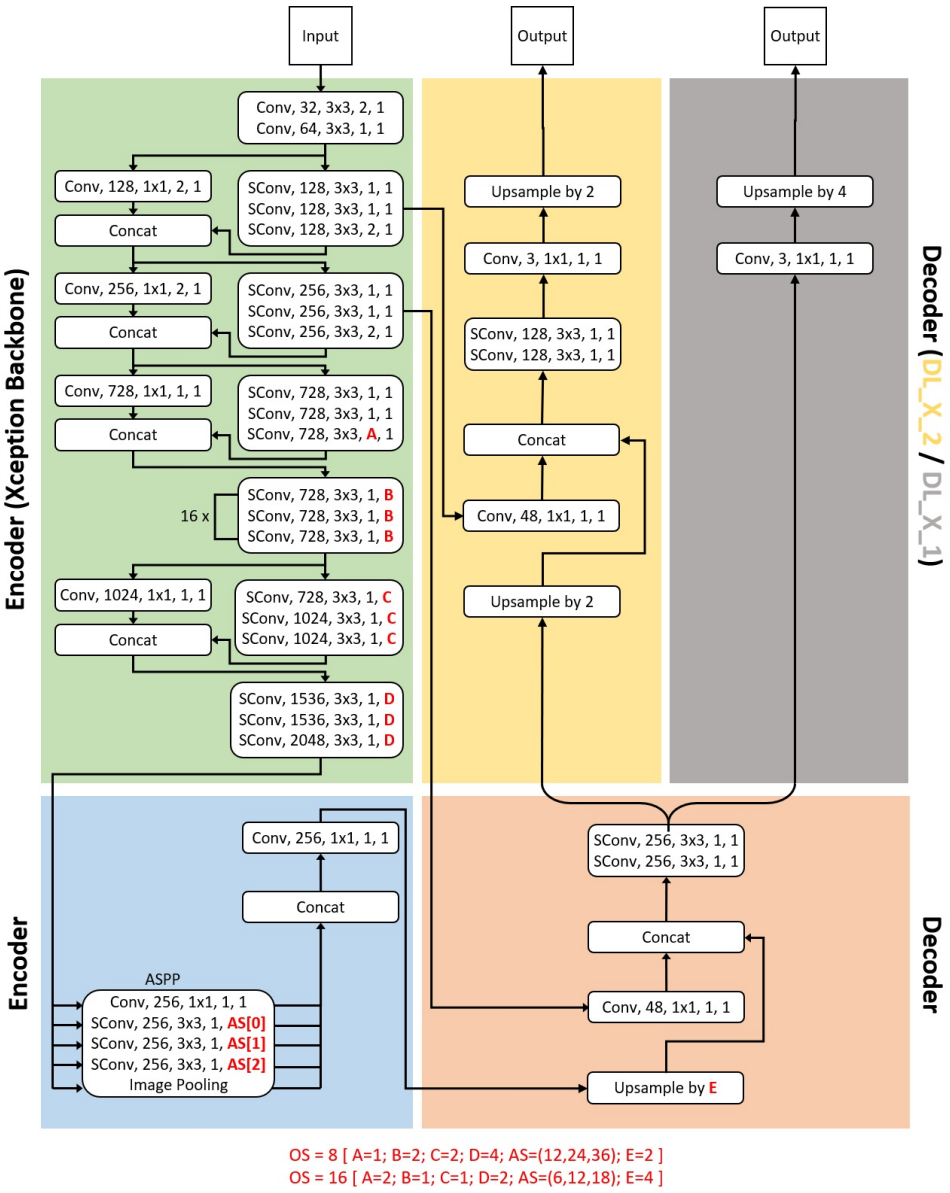
Embora os mapas de ativação do *bottleneck* constituam uma representação compacta de características (*features*) da imagem de entrada, informações espaciais sobre a imagem são perdidas durante o processo de redução da resolução. Para a recuperação dessas informações, as chamadas *skip connections* (conexões de salto) são utilizadas para passar essas características do *encoder* para o *decoder*, como proposto no modelo U-Net (RONNEBERGER; FISCHER; BROX, 2015) (Figura 10).

Figura 10 – Arquitetura da U-Net



No modelo DeepLabv3+, a quantidade de compactação entre a imagem de entrada e o *bottleneck* pode ser ajustada alterando os *strides* (passos) de algumas das operações de convolução no *encoder*; ajustando o fator de *upsampling* da primeira interpolação bilinear no *encoder*; e ajustando as taxas de dilatação de algumas das convoluções dilatadas em conformidade, como mostrado na Figura 11.

Figura 11 – Variantes do DeepLabv3+. As descrições das camadas contêm: tipo de convolução (Conv para convolução regular; SConv para convolução separável em profundidade), número de filtros, tamanho do filtro, *stride*, taxa de dilatação.



Nota: Os rótulos numéricos à esquerda dos blocos convolucionais indicam uma sequência de blocos idênticos. Os parâmetros mostrados em vermelho são variados para obter diferentes *output strides* (OS) para o *encoder*. A arquitetura do *decoder* original é representada pela área cinza, denotada DL_X_1, a área amarela representa a arquitetura alternativa DL_X_2 proposta, a letra X nesta notação é substituída posteriormente pelo *output stride* da variante.

Neste trabalho será avaliado o modelo DeepLabv3+ com diferentes *output strides* OS do *encoder*, sendo eles 8 e 16, que são obtidos alterando os valores dos parâmetros mostrados em fonte vermelha na Figura 11.

Além disso, foi proposta uma modificação da arquitetura original, na qual foi incluída uma

segunda *skip connection* entre o *encoder* e os módulos do *decoder* (mostrado na área amarela na Figura 11, observe que a área cinza representa a arquitetura do *decoder* original). Basicamente, a última interpolação bilinear no *decoder*, que realiza um *upsample* por um fator de 4, foi dividida em duas interpolações bilineares por um fator de 2. A nova *skip connection* foi então colocada entre essas interpolações de forma a recuperar mais detalhes espaciais nas características (*features*) do *encoder*, na tentativa de melhorar os resultados da segmentação semântica, especialmente nas bordas dos objetos segmentados.

Portanto, foram criadas quatro implementações alternativas do modelo DeepLabv3+. Duas variantes seguem a arquitetura original, com OS de 8 e 16 (com apenas uma *skip connection*). Daqui em diante, essas variantes serão denotadas como DL_8_1 e DL_16_1, respectivamente. As duas outras variantes representam a modificação proposta neste trabalho, ou seja, a inclusão de uma segunda *skip connection* no *decoder*, com OS de 8 e 16 para o *encoder*. Essas variantes serão denotadas como DL_8_2 e DL_16_2. Nos experimentos, foram avaliadas cada uma das quatro variantes para avaliar suas acurácias relativas.

3.2 **Descrição das bases de dados**

As bases de dados consistem em conjuntos de pares de imagens correlacionadas: uma imagem adquirida por microscopia de luz refletida e sua imagem de referência correspondente. A microscopia correlativa foi utilizada para obter imagens adequadamente registradas a partir de um microscópio de luz refletida e um MEV. As imagens óticas foram adquiridas na quantização de cor RGB com 24 bits, e as imagens de BSE do MEV são imagens, monocromáticas (com uma única banda), de 8 bits. Posteriormente, as imagens de BSE foram processadas para compor as imagens binárias de referência.

Neste estudo, não foram considerados erros decorrentes da microscopia correlativa (colocalização de campos e registro de imagens) ou devido ao processamento de imagens (delineamento e limiarização), assim como diferenças devido à natureza distinta das técnicas de imageamento empregadas. Por exemplo, imagens de microscopia de luz refletida e de MEV são geradas em diferentes profundidades na amostra polida, portanto, são capazes de mostrar características diferentes da amostra. Desse modo, cada imagem de referência é considerada uma imagem

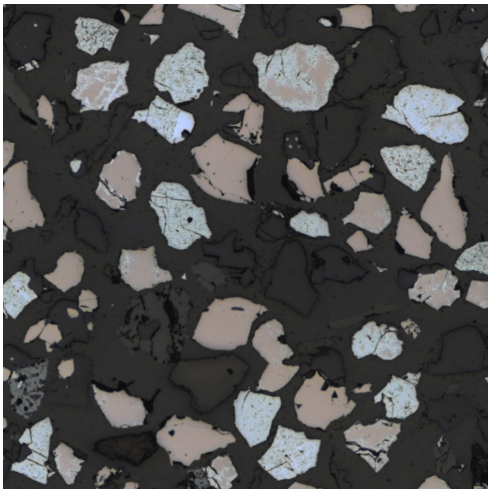
rotulada corretamente.

3.2.1 Base de dados Fe19

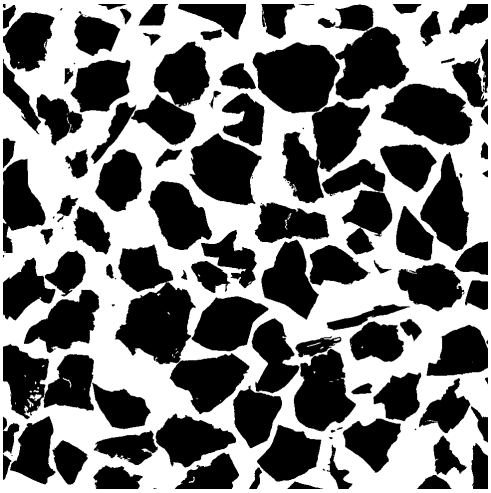
Esta base de dados contém imagens de uma amostra de minério de ferro do depósito de Serrote do Breu. A mineralogia deste depósito de minério de ferro compreende quartzo, magnetita, hematita, anfibólio (hastingsita), albita, biotita, calcita, goethita e caulinita, com um pouco de clorita. A magnetita é o principal mineral de ferro, seguida pela hematita e muito pouca goethita. A fração $+212 - 300$ μm foi embutida a frio em resina epóxi e posteriormente desbastada e polida. Foram imageados 19 campos em um microscópio de luz refletida com uma lente objetiva de 5X (NA 0,13) e em um MEV. A seguir, eles foram registrados, resultando em imagens de 972 x 972 pixels com resolução de 2,17 $\mu m/pixel$. A descrição desta amostra e seu procedimento de aquisição e registro de imagens podem ser encontrados em (GOMES; VASQUES; NEUMANN, 2018).

As imagens de MEV foram processadas para compor as imagens de referência usando o software de código aberto Fiji/ImageJ (SCHINDELIN et al., 2012). Primeiro, eles foram pré-processados com um filtro de delineação (raio = 1,5 e limiar = 40) (RP; CL, 1993) implementado como uma macro do Fiji, conforme descrito em (GOMES, 2018). O delineamento (realce de borda) converte as mudanças graduais no nível de cinza entre fases em mudanças abruptas, fazendo com que a transição de uma fase para a outra seja realizada em um passo de um único pixel. A seguir, as imagens delineadas foram limiarizadas: pixels com níveis de cinza entre 0 e 80 foram segmentados como resina (branco) e pixels com níveis de cinza acima de 80 foram definidos como partículas de minério (preto). A Figura 12 mostra um exemplo de um par de imagens do conjunto Fe19.

Figura 12 – Exemplo de imagem ótica e de referência do conjunto Fe19



(a) Imagem ótica

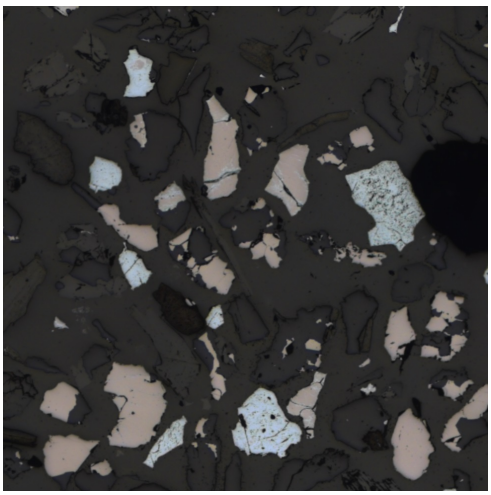


(b) Imagem de referência

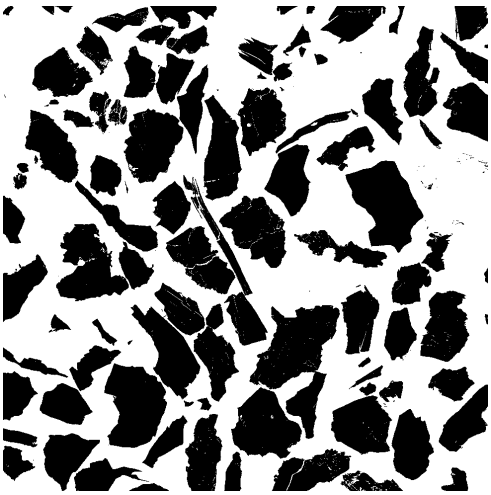
3.2.2 Base de dados Fe120

As imagens dessa base de dados vieram da mesma amostra da base de dados Fe 19, porém foram adquiridas em uma seção diferente, em um experimento com as mesmas condições. Assim, 120 campos foram imageados em um microscópio de luz refletida com uma lente objetiva de 5X (NA 0,13) e em um MEV. A seguir, eles foram registrados, resultando em imagens de 976 x 976 pixels com resolução de 2,17 μm/pixel. Então, como descrito acima, as imagens do MEV foram limiarizadas para compor as imagens de referência. A Figura 13 mostra um exemplo de um par de imagens do conjunto Fe120.

Figura 13 – Exemplo de imagem ótica e de referência do conjunto Fe120



(a) Imagem ótica



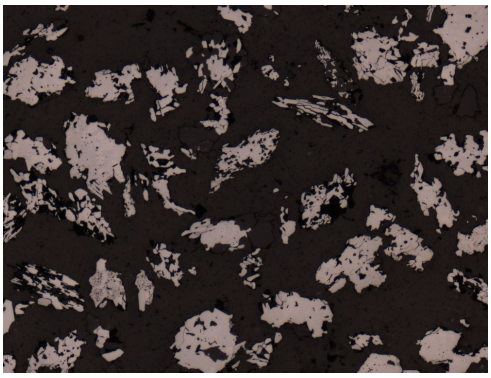
(b) Imagem de referência

3.2.3 Base de dados FeM

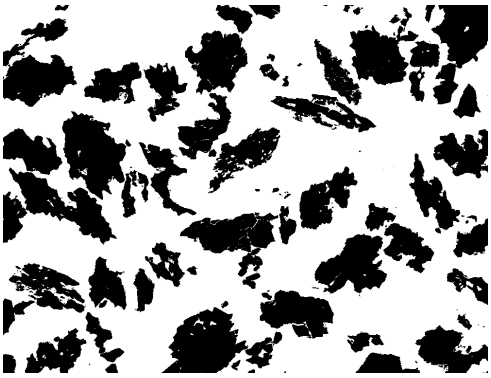
As imagens desta base de dados são de um minério de ferro itabirítico do Quadrilátero Ferrífero (Brasil) composto principalmente de hematita e quartzo, com poucas magnetita e goethita, que foi classificado e concentrado com um líquido denso. Assim, a fração $-149+105$ μm com densidade maior que 3,2 foi embutida a frio com resina epóxi e posteriormente desbastada e polida. 81 campos foram imageados em um microscópio de luz refletida com uma lente objetiva de 10X (NA 0,20) e em um MEV. A seguir, eles foram registrados, resultando em imagens de 999 x 756 pixels com resolução de 1,05 $\mu m/pixel$. A descrição desta amostra e seu procedimento de aquisição e registro de imagens podem ser encontrados em (GOMES; PACIORNIK, 2008b).

As imagens do MEV foram limiarizadas para compor as imagens de referência: pixels com níveis de cinza entre 0 e 70 foram segmentados como resina (branco) e pixels com níveis de cinza acima de 70 foram definidos como partículas de minério (preto). A Figura 14 mostra um exemplo de um par de imagens do conjunto FeM.

Figura 14 – Exemplo de imagem ótica e de referência do conjunto FeM



(a) Imagem ótica



(b) Imagem de referência

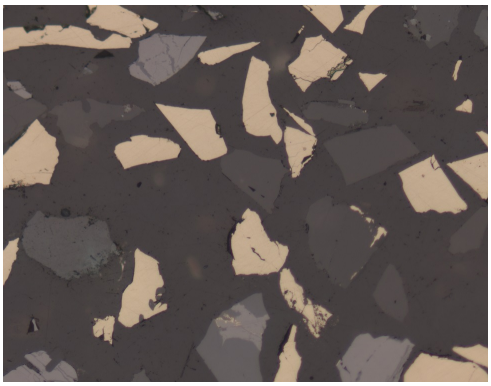
3.2.4 Base de dados Cu

Esta base de dados contém imagens de um minério de cobre de Yauri Cusco (Peru) com uma mineralogia complexa, composta principalmente por sulfetos, óxidos, silicatos e cobre nativo. O minério foi classificado em termos de tamanho de partícula. A fração $+74-100$ μm foi embutida a frio com resina epóxi e, posteriormente, desbastada e polida. Foram imageados 121 campos em um microscópio de luz refletida com uma lente objetiva de 20X (NA 0,40) e

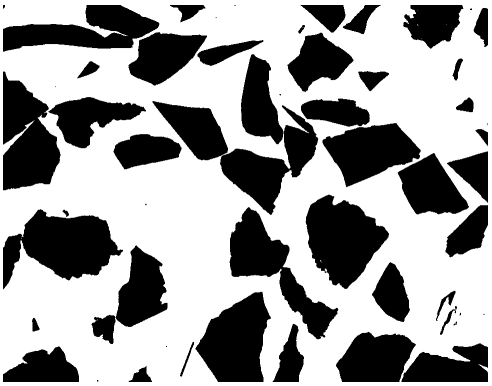
em um MEV. Em seguida, foram registrados, resultando em imagens de 1017 x 753 pixels com resolução de 0,53 µm/pixel. A descrição completa desta amostra e seu procedimento de imagem podem ser encontrados em (GOMES; PACIORNIK, 2008a).

As imagens do MEV foram limiarizadas para compor as imagens de referência: pixels com níveis de cinza entre 0 e 30 foram segmentados como resina (branco) e pixels com níveis de cinza acima de 30 foram definidos como partículas de minério (preto). A Figura 15 mostra um exemplo de um par de imagens do conjunto Cu.

Figura 15 – Exemplo de imagem ótica e de referência do conjunto Cu



(a) Imagem ótica



(b) Imagem de referência

3.3 **Bibliotecas**

Nesta subseção são apresentadas as bibliotecas utilizadas no trabalho. São elas:

- **Keras** - API de redes neurais de alto nível, escrita em Python e capaz de rodar sobre TensorFlow, CNTK ou Theano.
- **Numpy** - Pacote fundamental para a computação científica com Python. Ele contém entre outras coisas: matrizes N-dimensionais; recursos úteis de álgebra linear, transformação de Fourier e números aleatórios; entre outros.
- **OpenCV** - Biblioteca multiplataforma para o desenvolvimento de aplicativos na área de Visão Computacional. Possui módulos de Processamento de Imagens e Video I/O, Estrutura de dados, Álgebra Linear, além de mais de 350 algoritmos de Visão Computacional como: Filtros de imagem, calibração de câmera, reconhecimento de objetos, análise estrutural e outros.

- **Scikit** - Ferramentas simples e eficientes para mineração e análise de dados. Construído em NumPy, SciPy e matplotlib. Responsável pela geração das métricas da rede.
- **TensorFlow** - Plataforma de código aberto de ponta a ponta para aprendizado de máquina.
- **CUDA** - Plataforma de computação paralela, criada para que desenvolvedores possam usar de forma mais precisa e livre o alto potencial de processamento paralelo proporcionado por uma placa gráfica.

4 EXPERIMENTOS

Neste trabalho foram realizados quatro conjuntos de experimentos: (i) primeiro foram avaliadas as variantes DeepLabv3+ (DL_16_1, DL_8_1, DL_16_2 e DL_8_2) usando o conjunto de dados Fe19; (ii) então, foi selecionada a melhor variante (considerando o conjunto de dados Fe19) e foi avaliado o desempenho desse modelo usando os outros conjuntos de dados (Fe120, FeM e Cu) para treinamento e teste; (iii) na sequência, foram realizados experimentos de validação cruzada, nos quais a variante do modelo selecionada em (i) é treinada usando um conjunto de dados e testada usando todos os outros conjuntos de dados; (iv) por fim, foi realizado um experimento de treinamento com as bases Fe19 e Cu combinadas utilizando a variante do modelo selecionada em (i) e testada em todos os quatro conjuntos de dados.

Na próxima seção, são descritas as métricas usadas para avaliar o desempenho da segmentação semântica da abordagem proposta. Em seguida, é descrita a preparação das bases de dados e dos experimentos. Por fim, é detalhada a parametrização dos modelos de aprendizagem profunda.

4.1 Métricas utilizadas na avaliação dos resultados

Os modelos foram analisados com base na segmentação semântica das imagens. Desse modo cada um dos pixels da imagem de entrada da rede será classificado em uma das duas classes: resina ou minério. Essas imagens segmentadas são comparadas com suas respectivas imagens de referência e seus pixels são rotulados como: True Positive (TP); False Positive (FP); True Negative (TN); False Negative (FN). Isso gera uma matriz de confusão, um método comumente utilizado para medir a performance de um classificador, conforme mostra a Figura 16.

Figura 16 – Matriz de Confusão

ACTUAL	PREDICTED		
		NEGATIVE	POSITIVE
	NEGATIVE	TRUE NEGATIVE	FALSE POSITIVE
	POSITIVE	FALSE NEGATIVE	TRUE POSITIVE

Com isso nós temos:

- True Positive – Todo pixel rotulado corretamente como resina
- False Positive – Todo pixel que deveria ser rotulado como resina, porém foi rotulado como minério
- True Negative – Todo pixel rotulado corretamente como minério
- False Negative – Todo pixel que deveria ser rotulado como minério, porém foi rotulado como resina

A partir dessa rotulação é possível calcular os valores de *Overall Accuracy*, *Precision*, *Recall* e *F1 Score* com base nas formulas a seguir:

Overall Accuracy é uma métrica de avaliação que permite medir o número total de previsões que um modelo acerta. No entanto, ele não fornece informações detalhadas sobre sua aplicação ao problema. Para tal são usadas as métricas de *Precision* e *Recall*.

$$OverallAccuracy = \frac{TP + TN}{TN + FN + TP + FP} \tag{1}$$

Precision avalia a precisão de um modelo na previsão de rótulos positivos. Do número de vezes que um modelo previu positivo, com que frequência ele estava correto. É a porcentagem dos resultados que são relevantes.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

Recall calcula a porcentagem de positivos verdadeiros que um modelo identificou corretamente (True Positive).

$$Recall = \frac{TP}{FN + TP} \tag{3}$$

F1 Score é uma medida geral da precisão de um modelo que combina *Precision* e *Recall*. Ou seja, um bom *F1 Score* significa que o modelo apresenta poucos falsos positivos e falsos negativos.

$$F1Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{4}$$

4.2 Preparação da Base de Dados e dos Experimentos

O número de imagens respectivamente disponíveis para treinamento, validação e teste para cada base de dados é apresentado na Tabela 1. Cada linha representa a quantidade de imagens para cada base de dados. A primeira coluna indica o número total de imagens selecionadas, enquanto as colunas seguintes indicam o número de imagens atribuídas aos conjuntos de treinamento, validação e teste, respectivamente.

Tabela 1 – Quantidade de imagens utilizadas nos experimentos.

	Total de Imagens	**Imagens de Treinamento**	**Imagens de Validação**	**Imagens de Teste**
Fe19	19	10	5	4
Fe120	19	10	5	4
FeM	39	23	12	4
Cu	39	23	12	4

Conforme mostrado na seção 3.2, foram gerados diferentes conjuntos de imagens para cada grupo de minério. As imagens de minério são compostas por 3 bandas (R,G,B), enquanto as imagens de referências possuem uma única banda e são rotuladas em duas classes: minério (preto) e resina (branco). Todas as imagens foram normalizadas em um intervalo de [0, 1]. Para cada uma foram extraídos recortes (*patches*) de 512 x 512 pixels. Cada recorte é formado por um

pedaço da imagem original empilhada com um pedaço com as mesmas dimensões e na mesma posição da imagem de referência correspondente, gerando, assim, uma matriz de dimensões 512 x 512 x 4. Por fim esses recortes foram salvos em disco no formato de *Numpy Array (.npy)*.

Os recortes (*patches*) da imagem foram extraídos usando um procedimento de janela móvel. Para os conjuntos de treinamento e validação, o *stride* (passo) da janela foi de 256 pixels nas direções horizontal e vertical. Para o conjunto de teste, o *stride* foi de 512 pixels, para que os recortes resultantes não se sobreponham. Além disso, os conjuntos de treinamento e validação foram submetidos a *data augmentation* com o objetivo de aumentar a quantidade de amostras disponíveis, visando melhorar o desempenho de classificação e a capacidade de generalização dos modelos de aprendizagem profunda. Para tanto, os recortes de treinamento e validação foram rotacionados em 90º, 180º e 270º.

A fim de padronizar os experimentos, a partir de cada conjunto de dados de minério, o mesmo número de recortes (*patches*) foi selecionado (aleatoriamente) a partir do resultado do procedimento descrito acima, resultando em: 360 recortes para treinamento, 180 recortes para validação e 16 recortes para teste.

4.3 **Parametrização da Rede**

4.3.1 *Output Stride*

Um dos parâmetros principais para o desenvolvimento deste trabalho. O *output stride* explica a razão entre o tamanho da imagem de entrada e o tamanho do mapa de ativação de saída. Ele define quanta decimação de sinal o vetor de entrada sofre ao passar pela rede. O DeepLab trabalha com duas configurações de *output stride*, 8 e 16.

Para um *output stride* de 16, uma imagem de tamanho 512 x 512 produz um mapa de ativação com dimensões 16 vezes menores, ou seja 32 x 32. Analogamente, para um *output stride* de 8 a mesma imagem de 512 x 512 produzirá um mapa de ativação com dimensões 8 vezes menores, ou seja 64 x 64.

Além disso, o modelo DeepLab também debate os efeitos de diferentes *output strides* nos modelos de segmentação. Em suma, os modelos com menor *output stride* - menos decimação do sinal - tendem a produzir resultados de segmentação mais finos. No entanto, os modelos de

treinamento com menor *output stride* exigem mais tempo de treinamento, uma vez que reduzem menos o tamanho da imagem de entrada até o *bottleneck*.

4.3.2 Função de Ativação

Em redes neurais, a função de ativação de um neurônio define a saída desse neurônio dada uma entrada ou conjunto de entradas. Também é conhecida como Função de Transferência, uma vez que é responsável por mapear as entradas de um neurônio aos neurônios subsequentes.

Durante todos os experimentos foi utilizada a função de ativação *Sigmoid*. Esta função é comumente usada na camada de saída de um classificador binário, onde o resultado deverá ser 0 ou 1, uma vez que o intervalo dessa função sempre ficará entre $[0,1]$. Desse modo o resultado pode ser facilmente previsto como 1, caso o valor da função seja maior que $0,5$, ou 0, caso contrário. Ela é dada por:

$$Sigmoid(x) = \frac{1}{1+e^{-x}} \tag{5}$$

Onde:

$x =$ valor de entrada do neurônio

Além disso a função é diferenciável, o que significa que é possível encontrar a inclinação da curva em quaisquer dois pontos. Isso é importante por conta do *backpropagation*, uma vez que os cálculos são realizados através da regra da cadeia aplicada à rede. O algoritmo de *backpropagation* consiste em duas fases:

- *Forward pass* - as entradas são passadas através da rede e as previsões de saída são obtidas.
- *Backward pass* - o gradiente da função de perda na camada final da rede é calculado e esse gradiente é usado para aplicar recursivamente a regra da cadeia.

Desse modo a propriedade de diferenciação da função é necessária para que os pesos da rede possam ser atualizados.

4.3.3 Função de Perda

As redes neurais são treinadas usando métodos iterativos de otimização. Como parte do algoritmo de otimização, o erro para o estado atual do modelo deve ser continuamente estimado. Isso requer a escolha de uma função de erro, convencionalmente chamada de função de perda, que pode ser usada para estimar a perda do modelo de forma que os pesos possam ser atualizados, tentado minimizar a perda nas próximas avaliações.

Binary Cross-Entropy é a função de perda padrão usada em problemas de classificação binária onde os valores desejados estão no conjunto $[0,1]$. A função calcula um valor que resume a diferença média entre as distribuições de probabilidade real e prevista para a classe de previsão. O valor da função aumenta conforme a probabilidade prevista diverge do rótulo real. Ou seja, um modelo perfeito deve apresentar um valor de perda igual a 0. A função de perda *Binary Cross-Entropy* é dada por:

$$L = -(y \times log(p) + (1-y) \times log(1-p)) \tag{6}$$

Onde:

$y =$ indicador binário (0 ou 1) se o rótulo da classe c é a classificação correta para observação o

$p =$ observação o da probabilidade prevista é da classe c

4.3.4 Otimizador

Otimizadores são algoritmos ou métodos usados para alterar os atributos de uma rede neural, como pesos e taxa de aprendizado, a fim de reduzir a perda e fornecer os resultados mais precisos possíveis.

O otimizador usado durantes os experimentos foi o Adam. Ele é uma extensão do método de *Stochastic Gradient Descent (SGD)* que se baseia na estimativa adaptativa de momentos de primeira e segunda ordem. O método é computacionalmente eficiente, tem poucos requisitos de memória, invariante para o reescalonamento diagonal dos gradientes e é adequado para problemas que são grandes em termos de dados/parâmetros, de acordo com (KINGMA; BA, 2014).

A parametrização do otimizador foi feita seguindo as recomendações sugeridas em (KINGMA; BA, 2014). Sendo assim o otimizador foi configurado da seguinte maneira:

```
tf.keras.optimizers.Adam(
    learning_rate=0.001,
    β1 = 0.9,
    β2 = 0.999,
    ε = 1e−07,
)
```

Onde:

- learning rate - proporção em que os pesos são atualizadas. Valores maiores resultam em um aprendizado inicial mais rápido antes que a taxa seja atualizada. Valores menores retardam o aprendizado durante o treinamento.
- β_1 - taxa de decaimento exponencial para as estimativas de primeiro momento.
- β_2 - taxa de decaimento exponencial para as estimativas de segundo momento.
- ε - um número muito pequeno para evitar qualquer divisão por zero na implementação.

4.3.5 Outros Parâmetros

Os pesos da rede foram inicializados aleatoriamente seguindo a técnica Glorot Uniform de acordo com (GLOROT; BENGIO, 2010). Além disso, os três primeiros experimentos foram feitos utilizando 25 épocas de treinamento. Este valor foi escolhido com base em avaliações prévias usando a técnica de *Early Stop*. O *Early Stop* é uma forma de regularização usada para evitar *overfitting*[1] ao treinar um modelo com um método iterativo. Quando o modelo não apresenta melhora nos seus valores de *accuracy* e *loss* por uma certa quantidade (predefinida) de épocas, o treinamento é interrompido antecipadamente e seus pesos são salvos.

O hardware usado nos testes continha uma GPU NVidia Quadro P5000. Esta GPU possui 2560 núcleos CUDA e uma memória de vídeo GDDR5X de 16 GB. O *batch size* usado, ou seja,

[1] Fenômeno que ocorre quando o modelo se adaptou muito bem aos dados com os quais está sendo treinado; porém, não generaliza bem para novos dados.

o número de *patches* de imagem processados simultaneamente foi 1, uma vez que era o máximo que a GPU poderia processar simultaneamente.

5 RESULTADOS E DISCUSSÃO

Nesta seção são apresentados e discutidos os resultados dos experimentos realizados neste trabalho. Inicialmente é feita a comparação dos desempenhos das variantes do modelo DeepLabv3+ sobre o conjunto de dados Fe19. Em seguida, é avaliado o desempenho da melhor variante do modelo, treinando e testando-a com dados de cada um dos outros conjuntos de dados individualmente. Após isso, são apresentados os resultados dos experimentos de validação cruzada, nos quais o modelo é treinado com dados de um conjunto de dados e testado com dados de todos os outros conjuntos de dados. Finalmente, são apresentados os resultados dos experimentos de treinamento com as bases Fe19 e Cu combinadas.

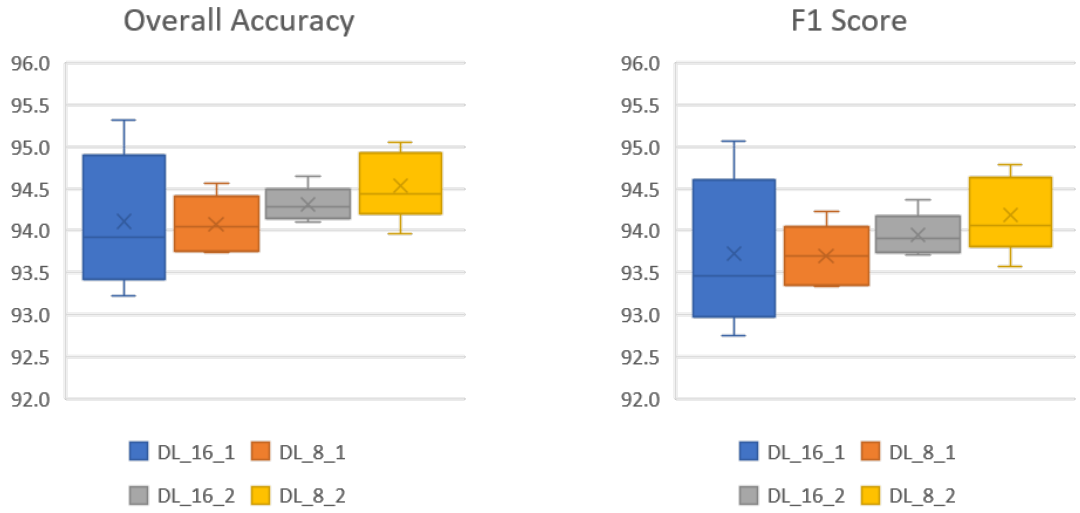
5.1 Comparação das variantes DeepLabv3+

A Tabela 2 apresenta os resultados em termos de *Overall Accuracy* e *F1 Score* obtidos para os experimentos de segmentação semântica com o conjunto de dados Fe19 com as variantes do modelo DeepLabv3+. A tabela mostra a médias, medianas e desvios padrão de 5 rodadas de experimentos (treinamento e teste), cada um com uma inicialização aleatória diferente de pesos das camadas. Como pode ser visto, as variantes do modelo com a modificação na arquitetura proposta neste trabalho, ou seja, a inclusão de outra *skip connection* entre o *encoder* e o *decoder* (DL_16_2 e DL_8_2) apresentam uma ligeira, embora consistente melhora no desempenho em relação aos modelos com a arquitetura original (DL_16_1 e DL_8_1). A significância da melhoria foi validada com um teste de hipótese *t-test* feito no matlab. A Figura 17 mostra os gráficos de *Overall Accuracy* e *F1 Score* comparando as quatro variantes do modelos proposto. Nela é possível observar uma significativa melhora obtida pela arquitetura proposta quando comparada à original, principalmente em relação às médias e desvios padrão.

Tabela 2 – *Overall Accuracy* (%) e *F1 Score* (%) para 5 rodadas de execução para cada uma das 4 variações do modelo. Os melhores resultados são mostrados em negrito.

Modelo/Métrica	Overall Accuracy			F1 Score		
	Média	**Mediana**	**Desvio Padrão**	**Média**	**Mediana**	**Desvio Padrão**
DL_16_1	94,11	93,92	0,82	93,72	93,47	0,90
DL_8_1	94,07	94,04	0,35	93,70	93,70	0,37
DL_16_2	94,31	94,28	**0,21**	93,95	93,90	**0,25**
DL_8_2	**94,54**	**94,43**	0,42	**94,19**	**94,06**	0,46

Figura 17 – *Overall Accuracy* e *F1 Score* para as quatro variantes do modelo DeepLabv3+ propostas neste trabalho



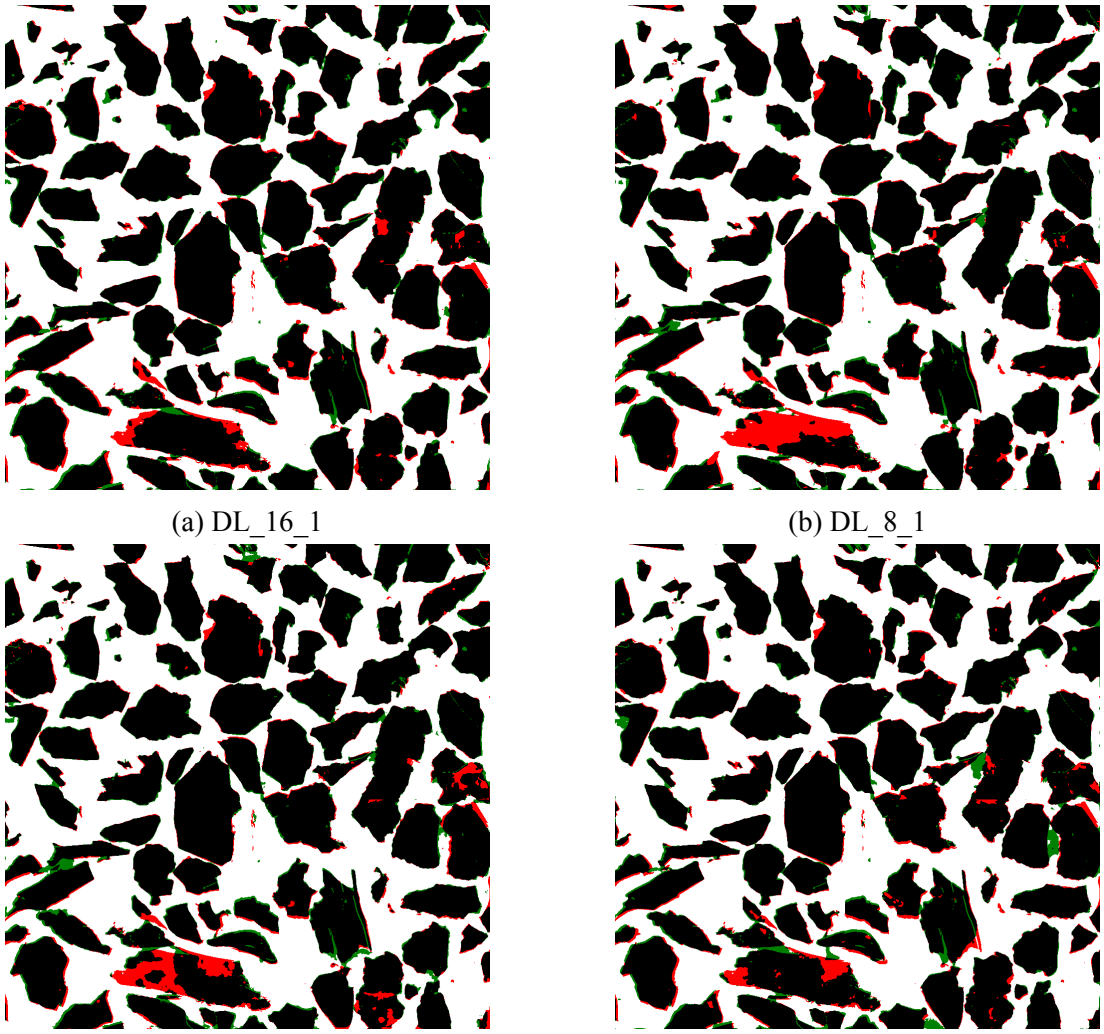
(a) *Overall Accuracy* (b) *F1 Score*

Também pode ser visto na Tabela 2 que os diferentes *output strides* produziram resultados semelhantes para as variantes que seguem a arquitetura original (DL_16_1 e DL_8_1). Considerando as variantes da arquitetura proposta (DL_16_2 e DL_8_2), os resultados também são semelhantes, com uma ligeira vantagem da variante DL_8_2. No entanto, os testes de hipótese para ambos os conjuntos de resultados não puderam rejeitar a hipótese nula de que suas médias são iguais, a um nível de significância de 5%.

A Figura 18 mostra os mapas de classificação de uma imagem do conjunto de dados Fe19 obtido com cada uma das 4 variantes do modelo. Nesses mapas, pixels corretamente classificados como resina (resina verdadeira) foram pintados de branco, pixels corretamente classificados como minério (minério verdadeiro) foram pintados de preto, pixels de minério classificados incorretamente como resina (resina falsa) foram pintados de vermelho, enquanto pixels de resina

classificada erroneamente como minério (minério falso) foram pintados de verde.

Figura 18 – Mapas de classificação de uma imagem do conjunto de dados Fe19. Código de cores: *branco é resina verdadeira, preto é minério verdadeiro, vermelho é resina falsa, verde é minério falso.*



(a) DL_16_1 (b) DL_8_1

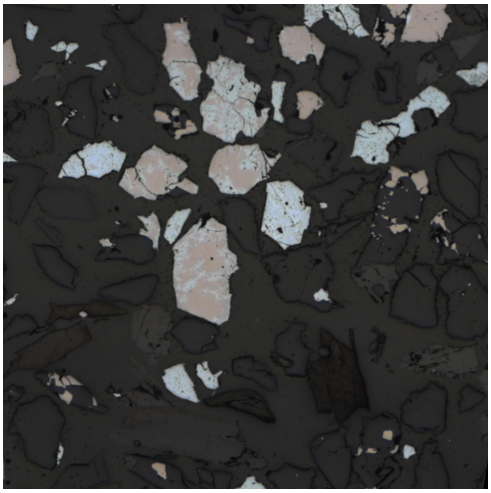
(c) DL_16_2 (d) DL_8_2

À primeira vista, pode-se observar que os mapas de classificação obtidos com todas as variantes são bastante satisfatórios, a maioria dos pixels da imagem foi classificadas corretamente (pixels de cor preta e branca), refletindo a *Overall Accuracy* e *F1 Score* próximas a 95%, apresentadas na Tabela 2. Mesmo os minerais não opacos foram em geral bem reconhecidos.

Os erros de classificação (pixels vermelhos e verdes) podem ser divididos em dois tipos: erros comuns a todos os mapas de classificação e erros relacionados a diferenças nos modelos. Os erros comuns são geralmente devidos à natureza distinta das técnicas usadas para obter as imagens de referência (MEV) e as imagens que foram classificadas (microscopia de luz refle-

tida). A Figura 19 mostra a imagem de microscopia de luz refletida e sua respectiva imagem de referência, associada aos mapas de classificação apresentados na Figura 18. A Figura 20 apresenta a imagem do BSE da qual foi obtida a imagem de referência (Figura 19b) e indica alguns erros de classificação.

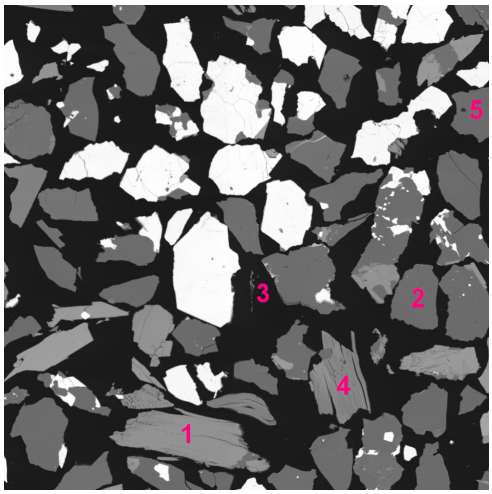
Figura 19 – Imagens de entrada do conjunto de dados Fe19 associado aos mapas de classificação mostrados na Figura 18.



(a) Imagem de microscopia de luz refletida

(b) Imagem de referência

Figura 20 – Imagem BSE da qual foi obtida a imagem de referência (Figura 19b), com indicação de alguns erros de classificação.



A maioria dos pixels erroneamente classificados nesta imagem pertence a uma partícula de mica na parte inferior, apontada com o número 1 na Figura 20. Pode-se sugerir que isso ocorreu porque essa partícula é muito semelhante à resina circundante, sendo difícil diferenciá-los até mesmo por uma pessoa. Os resultados da classificação dos pixels desta partícula variam

amplamente dependendo do modelo empregado, conforme mostrado na Figura 18.

Também existem erros de classificação nas bordas de algumas partículas de minério. Isso ocorre principalmente em partículas de quartzo que exibem um anel escuro nas imagens de microscopia de luz refletida, causado pelo relevo preferencial presente nessa seção polida. Este artefato de preparação de amostra não é visto nas imagens BSE correspondentes devido à maior profundidade de campo do MEV. Um exemplo desse fenômeno pode ser observado na partícula na posição 2 (conforme indicado na Figura 20), na imagem de microscopia de luz refletida (Figura 19a).

Próximo ao centro de todos os mapas de classificação, há um conjunto de pixels incorretamente rotulados, principalmente vermelhos (resina falsa), alinhados verticalmente. Na verdade, a região desses pixels na imagem da luz refletida é essencialmente resina, portanto os modelos os classificam como resina. No entanto, esses pixels foram rotulados como minério (preto) na imagem de referência, porque pequenas partículas de minério são visíveis nesta região na imagem BSE (Figura 20, posição 3). Eles provavelmente estão logo abaixo da superfície, em uma profundidade adequada para a emissão de BSE, mas imersos o suficiente para não refletirem a luz.

Outros erros referem-se a linhas verdes retas, como aquelas encontradas em uma partícula de mica na direção inferior direita. Na verdade, são fraturas em planos de clivagem visíveis na imagem BSE (Figura 20, posição 4) e, consequentemente, na imagem de referência, em branco, mas não na imagem de luz refletida. Assim, os modelos classificaram os pixels dessas fraturas como parte da partícula de minério em que estão embutidos como falso minério (verde).

Existe um erro, que embora possa ser raro, é interessante comentar aqui. Em uma partícula de minério próximo ao canto superior direito, há um ponto escuro na imagem BSE (Figura 20, posição 5), rotulado como resina (branco) na imagem de referência, que não está presente na imagem de luz refletida. Portanto, esta região foi classificada como minério, mas considerada falso minério (verde). Este ponto parece ser um dano na superfície do espécime causado pelo feixe de elétrons focalizado para microanálise EDS. Como a imagem de luz refletida foi adquirida antes da imagem MEV, este ponto está presente apenas na última.

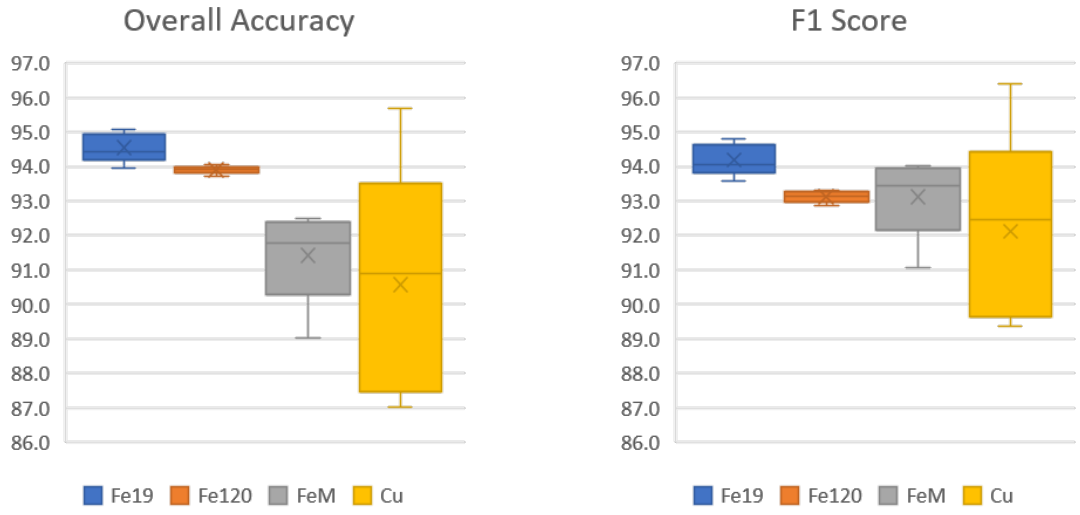
5.2 Segmentação semântica de outros conjuntos de dados

Considerando os resultados mostrados na seção anterior para o conjunto de dados Fe19, foram elaborados experimentos de segmentação semântica nos outros conjuntos de dados usando o modelo DL_8_2. A Tabela 3 apresenta os resultados obtidos em termos de *Overall Accuracy* e *F1 Score*. Ela mostra as médias, medianas e os desvios padrão de 5 rodadas de experimentos (treinamento e teste), cada um com uma inicialização diferente e aleatória de pesos das camadas. A Figura 21 mostra o *Overall Accuracy* e *F1 Score* do treinamento realizado com cada base de dados disponível utilizando a variante DL_8_2 do modelo DeepLabv3+.

Tabela 3 – *Overall Accuracy* (%) e *F1 Score* (%) para 5 rodadas de execução para a variante do modelo DL_8_2 aplicada a cada conjunto de dados.

Conjunto/Métrica	**Overall Accuracy**			**F1 Score**		
	Média	**Mediana**	**Desvio Padrão**	**Média**	**Mediana**	**Desvio Padrão**
Fe19	94,54	94,43	0,42	94,19	94,06	0,46
Fe120	93,90	93,91	0,12	93,12	93,12	0,16
FeM	91,43	91,77	1,39	93,13	93,44	1,20
Cu	90,56	90,90	3,41	92,12	92,45	2,78

Figura 21 – *Overall Accuracy* e *F1 Score* do treinamento de cada uma das bases de dados usando a variante DL_8_2 do modelo DeepLabv3+ proposta neste trabalho

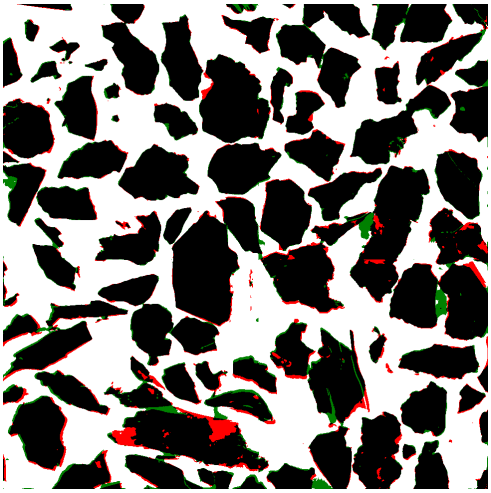


(a) *Overall Accuracy* (b) *F1 Score*

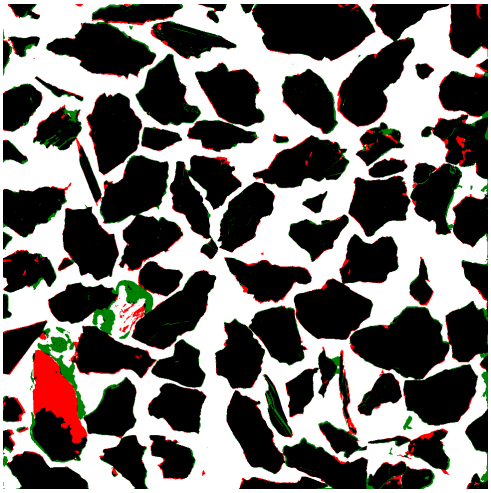
A Figura 22 mostra os mapas de classificação das imagens dos quatro conjuntos de dados utilizando a variante DL_8_2 do modelo. Nesses mapas, pixels corretamente classificados

como resina (resina verdadeira) foram pintados de branco, pixels corretamente classificados como minério (minério verdadeiro) foram pintados de preto, pixels de minério classificados incorretamente como resina (resina falsa) foram pintados de vermelho, enquanto pixels de resina classificada erroneamente como minério (minério falso) foram pintados de verde.

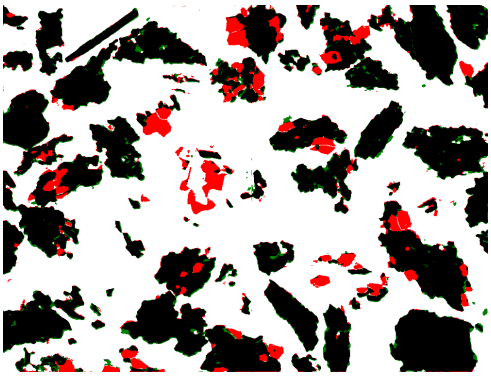
Figura 22 – Mapas de classificação das imagens dos quatro conjuntos de dados utilizando a variante DL_8_2 do modelo proposto. Código de cores: *branco é resina verdadeira, preto é minério verdadeiro, vermelho é resina falsa, verde é minério falso.*



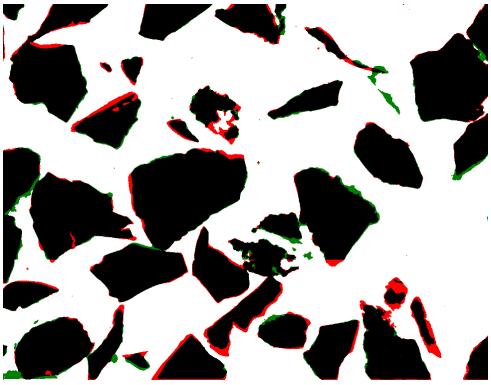
(a) Fe19



(b) Fe120



(c) FeM



(d) Cu

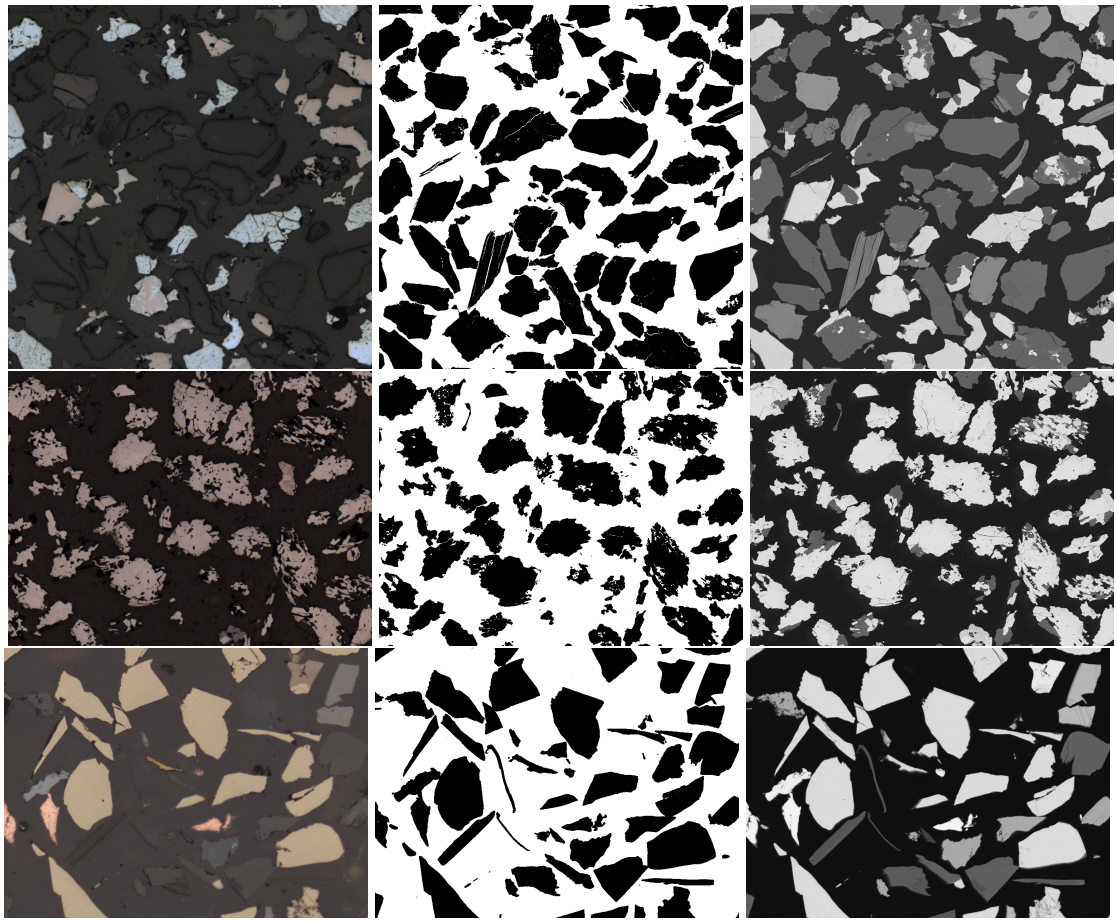
É possível observar que os mapas de classificação obtidos com todas as imagens dos diferentes conjuntos são bastante satisfatórios, a maioria dos pixels da imagem foi classificada corretamente (pixels de cor preta e branca), refletindo a *Overall Accuracy* e *F1 Score* acima dos 90% em todos os tipos de minério, conforme exibido na Tabela 3. Vale ressaltar o mapa de classificação da imagem de Cu (Figura 22d), nele é possível ver que praticamente todos os erros de classifação ocorreram nas regiões de borda dos minerais, diferente dos outros 3 conjuntos

que apresentam alguns erros em regiões centrais de algumas partículas de minério. O mapa de classificação da imagem FeM (Figura 22c), por sua vez, é o que apresenta a maior quantidade de erros de resina falsa, isso acontece, principalmente, em partículas menores ou em regiões onde o minério é mais poroso. Por fim, os mapas de classificação de Fe19 e Fe120 (Figuras 22a e 22b, respectivamente) são os mais constantes e os que apresentam menores erros de classificação. Os principais erros se dão em regiões bem específicas e comuns aos dois, ocorrendo quando a diferenciação, inclusive visual, entre minério e resina se torna mais difícil se refletindo na classificação do modelo.

Geralmente, os resultados da segmentação semântica para todos os conjuntos de dados mostraram magnitudes semelhantes, sempre entre 90 e 95% para a *Overall Accuracy* e *F1 Score*, o que parece indicar que o modelo DeepLabv3+, e particularmente a variante DL_8_2 proposta, foi capaz de lidar com os conjuntos de dados de diferentes minérios.

No entanto, embora a *Overall Accuracy* e os valores médios de *F1 Score* fossem semelhantes, suas dispersões variam amplamente. Por exemplo, o desvio padrão de *Overall Accuracy* variou de 0,12 a 3,41%, respectivamente para os conjuntos de dados Fe120 e Cu. Não está claro por que isso aconteceu, mas algumas hipóteses podem ser levantadas olhando as imagens. A Figura 23 apresenta um trio de imagens (luz refletida, referência e imagem MEV) de cada conjunto de dados Fe120, FeM e Cu.

Figura 23 – Exemplo de um trio de imagem correspondente (luz refletida, referência e imagem MEV) de cada conjunto de dados: Fe120 (a, b e c), FeM (d, e e f) e Cu (g, h e i).



Além de serem provenientes de diferentes minérios, as próprias imagens apresentam características distintas devido ao processamento do minério, ao procedimento de preparação da amostra e à configuração experimental empregada. Pode-se notar claramente na Figura 23 que as imagens dos conjuntos de dados FeM e Cu mostram mais resina de embutimento e, consequentemente, menos minério, em comparação com as imagens do conjunto de dados Fe120. Além disso, as partículas de minério do conjunto de dados de Cu parecem ser mais liberadas do que as partículas de Fe120 e FeM. Essas características tendem a aumentar a variabilidade entre as imagens de um determinado conjunto de dados e podem explicar as maiores dispersões nos resultados obtidos para o conjunto de dados FeM e, particularmente, para o conjunto de dados Cu.

5.3 **Experimentos de validação cruzada**

As tabelas 4 e 5 mostram a *Overall Accuracy* e *F1 Score* alcançadas em experimentos de validação cruzada. Em tais experimentos, a variante DL_8_2 do modelo Deeplabv3+ foi treinada com imagens de cada conjunto de dados e posteriormente avaliada em imagens de outros conjuntos de dados.

Tabela 4 – Medidas de *Overall Accuracy* (%) e *F1 Score* (%) para os experimentos de validação cruzada, treinando o modelo com os conjuntos de treinamento Fe19 e Fe120 e avaliando-os com os conjuntos de teste dos outros conjuntos de dados.

Conjunto de treino	**Fe19**			**Fe120**		
Conjunto de teste	**Fe120**	**FeM**	**Cu**	**Fe19**	**FeM**	**Cu**
Overall Accuracy	94,92	36,92	38,13	92,68	45,03	40,65
F1 Score	94,27	33,27	10,25	92,19	31,63	14,59

Tabela 5 – Medidas de *Overall Accuracy* (%) e *F1 Score* (%) para os experimentos de validação cruzada, treinando o modelo com os conjuntos de treinamento FeM e Cu e avaliando-os com os conjuntos de teste dos outros conjuntos de dados.

Conjunto de treino	**FeM**			**Cu**		
Conjunto de teste	**Fe19**	**Fe120**	**Cu**	**Fe19**	**Fe120**	**FeM**
Overall Accuracy	53,07	53,68	40,23	54,05	60,06	79,32
F1 Score	62,02	58,47	1,61	67,23	68,82	85,53

Os modelos treinados com imagens Fe19 e Fe120 apresentaram bons resultados na segmentação semântica das imagens entre si, mas os resultados foram ruins para os outros minérios (FeM e Cu). O modelo treinado com Fe19 atingiu 94,92% para *Overall Accuracy* e 94,27% para *F1 Score* na classificação das imagens do Fe120, enquanto o modelo treinado com Fe120 atingiu, respectivamente 92,68% e 92,19% para *Overall Accuracy* e *F1 Score* na classificação das imagens do Fe19. Em contraste, os resultados para a segmentação de imagens dos conjuntos de dados FeM e Cu foram sistematicamente abaixo de 50%.

Esses resultados parecem sugerir que a variante DL_8_2 do modelo generaliza bem no caso de imagens de uma mesma amostra de minério adquiridas de forma independente, mas com a mesma configuração experimental. Vale lembrar de que os conjuntos de dados Fe19 e Fe120

foram adquiridos de diferentes seções polidas da mesma amostra. Ao mesmo tempo, os resultados indicam que quando o modelo é treinado com imagens de um determinado minério, ele não é capaz de generalizar para diferentes minérios, nem mesmo para outro minério de ferro, como o associado ao conjunto de dados FeM.

Na verdade, as imagens do conjunto de dados FeM apresentam algumas diferenças perceptíveis quando comparadas às imagens de outros conjuntos de dados. FeM contém imagens de um concentrado de minério, ao contrário de outros conjuntos de dados, pois apresenta muito menos minerais de ganga, como quartzo e outros silicatos não opacos (tons escuros de cinza em imagens de MEV), como pode ser visto comparando as figuras 23c, 23f e 23i. As imagens óticas do conjunto de dados FeM também diferem das imagens dos conjuntos de dados Fe19 e Fe120 em termos de brilho, contraste e cor, porque foram adquiridas com diferentes sistemas de imagem. Isso é evidente ao comparar as figuras 19a (Fe19), 23a (Fe120) e 23d (FeM). Além disso, o concentrado de minério FeM é composto principalmente de hematita (mineral cinza claro na Figura 23d), enquanto, no minério de Fe19 e Fe120, magnetita (cinza rosado nas figuras 19a e 23a) é o principal mineral de ferro, mas também há uma quantidade considerável de hematita (cinza azulado nas Figuras 19a e 23a).

Porém, o modelo treinado com imagens Cu apresentou resultados promissores, principalmente na segmentação das imagens do conjunto de dados FeM: 79,32% e 85,53% para *Overall Accuracy* e *F1 Score*, respectivamente. De fato, o modelo treinado com o conjunto de dados Cu obteve os melhores resultados na segmentação de imagens de diferentes minérios. Possivelmente isso ocorreu porque as imagens de microscopia de luz refletida do conjunto de dados Cu são bastante coloridas, enquanto as de minérios de ferro, mesmo sendo imagens RGB 24 bits, apresentam basicamente cores com tonalidades próximas da escala de cinza. Essa variabilidade de cores pode ser o motivo da capacidade de generalização do modelo treinado com as imagens Cu.

5.4 **Treinamento com as bases Fe19 e Cu combinadas**

Para realizar esses experimentos os recortes (*patches*) dos conjuntos de treinamento e teste das bases Fe19 e Cu foram combinados. Desse modo os conjuntos de treinamento e validação

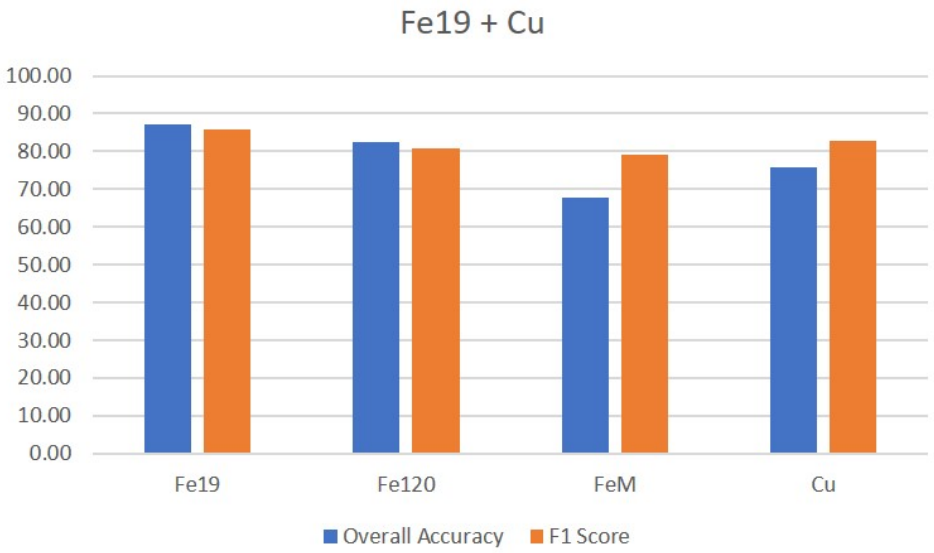
eram compostos por 720 (360 do conjunto Fe19 e 360 do conjunto Cu) e 360 recortes (180 do conjunto Fe19 e 180 do conjunto Cu), respectivamente. Por se tratar de um treinamento com um número muito maior de recortes, a quantidade de épocas utilizada também foi aumentada, passando de 25 para 40 épocas. Durante todo o experimento foi utilizada a variante DL_8_2, além disso, todas as demais configurações do modelo também se mantiveram iguais às anteriores.

As imagens de Fe19 e Cu foram escolhidas tendo como base os experimentos anteriores. A base de dados Fe19 obteve resultados altos e seu modelo treinado conseguiu generalizar muito bem as imagens da base de dados Fe120. Já a base de dados Cu foi escolhida tendo em vista que seu modelo foi o que melhor classificou as imagens da base FeM durante os experimentos de validação cruzada. Desse modo, o objetivo durante esse último experimento foi avaliar o desempenho do DeepLabv3+, em específico, a variante DL_8_2, em um treinamento com imagens de diferentes domínios combinadas. A Tabela 6 e a Figura 24 apresentam os resultados obtidos em termos de *Overall Accuracy* e *F1 Score*.

Tabela 6 – Medidas de *Overall Accuracy* (%) e *F1 Score* (%) para o experimento de treinamento das bases Fe19 e Cu combinadas, avaliando-o com todos os conjuntos de teste.

Métrica / Inferência	**Fe19**	**Fe120**	**FeM**	**Cu**
Overall Accuracy	87,25	82,62	67,84	75,67
F1 Score	85,67	80,64	79,08	82,96

Figura 24 – Overall Accuracy e F1 Score do treinamento combinado das bases de dados Fe19 e Cu, usando a variante DL_8_2 do modelo DeepLabv3+.



É possível observar a partir da tabela 6 que todos os resultados foram superiores a 50%, mostrando que o modelo treinado com as duas bases combinadas tem uma precisão maior que um classificador aleatório. Também é possível observar que as bases Fe19 e 120 mantiveram os valores de *Overall Accuracy* e *F1 Score* próximos, mostrando que o modelo continua lidando bem com imagens de um mesmo domínios. A base de dados Cu obteve resultados bem próximos aos das bases Fe19 e 120, principalmente em relação ao *F1 Score*. Além disso foi possível comprovar que a base Cu consegue classificar bem as imagens de FeM, obtendo resultados próximos de 70% em *Overall Accuracy* e próximos de 80% em *F1 Score*. Entretanto, esses resultados ainda estão longe dos obtidos nos experimentos das seções 5.1 e 5.2 e necessitam de um estudo mais aprofundado.

CONCLUSÃO

Este trabalho apresentou uma abordagem de segmentação semântica baseada em aprendizagem profunda para a discriminação de minério e resina em imagens de microscopia ótica. A abordagem foi proposta para solucionar um problema clássico de classificação de microscopia de luz refletida associado à similaridade espectral entre o quartzo, bem como outros minerais de ganga não opacos, e a resina de incorporação.

A abordagem é baseada em um modelo específico de aprendizagem profunda, o Deeplabv3+. Neste trabalho foram avaliadas diferentes variantes do modelo, algumas das quais incluindo uma extensão de sua arquitetura. As modificações propostas obtiveram sucesso na obtenção de melhores acurácias em comparação às alcançadas com o modelo original, e produziram melhorias notáveis no delineamento dos contornos das partículas minerais. Como as fronteiras são as principais regiões de incerteza para a tarefa de segmentação, isso pode ser considerado um resultado qualitativo significativo.

O desempenho das variantes do modelo de aprendizagem profunda foi comparado usando o primeiro dos quatro conjuntos de dados explorados nos experimentos, o conjunto de dados Fe19, e a variante com uma *skip connection* adicional e *output stride* de 8 superou suas contrapartes, alcançando precisões acima de 94% (tanto em termos de *Overall Accuracy* e *F1 Score*). A melhor variante do modelo foi então treinada e avaliada usando os outros conjuntos de dados (avaliações intra grupo), e as precisões em todos os conjuntos de dados adicionais foram sistematicamente acima de 90% (em termos de *Overall Accuracy* e *F1 Score*).

Também foram realizadas avaliações entre grupos, nas quais o modelo treinado com uma base de dados foi avaliado em cada uma das outras bases de dados. Os resultados dos experimentos entre grupos apresentaram diversos níveis de precisão. O par de conjuntos de dados mais semelhantes, Fe19 e Fe120, que foram produzidos com diferentes seções polidas do mesmo espécime, mostraram resultados satisfatórios (recíprocos) entre os grupos. Os valores de precisão obtidos foram semelhantes aos seus respectivos resultados dentro do grupo. Para a maioria das outras combinações de conjuntos de dados, no entanto, o modelo apresentou precisões menores do que um classificador aleatório. No entanto, o modelo treinado com o conjunto de dados Cu produziu a maior precisão geral de validação cruzada, bem como uma generalização surpreen-

dentemente justa para o conjunto de dados FeM.

Considerando os conjuntos de dados avaliados como diferentes domínios de imagem, os resultados indicam que o modelo de aprendizagem profunda proposto generaliza bem apenas para domínios semelhantes, ou seja, Fe19 e Fe120, que contêm imagens obtidas de diferentes seções transversais do mesmo minério, adquiridas com o mesmo sistema de imageamento. É uma noção comum que quando a dissimilaridade entre domínios de imagem é maior, um modelo treinado com amostras do domínio de origem terá um desempenho ruim na classificação de uma amostra do domínio alvo. No caso das bases de dados selecionados, a disparidade entre domínios parece estar associada principalmente aos minerais presentes, mas também à preparação de amostras e sistemas de imageamento. Portanto, o estabelecimento de protocolos padronizados para preparação de amostras e aquisição de imagens deve aumentar a similaridade entre domínios e, eventualmente, melhorar a generalização de um modelo para diferentes amostras de um determinado minério.

Por fim, foi realizado um experimento de treinamento combinando as bases de dados Fe19 e Cu, onde foram unidos os conjuntos de treinamento e validação, e, em seguida, foi testado em todos as quatro bases de dados disponíveis. Nesse experimento foi possível notar uma significativa melhora na capacidade de generalização do modelo quando comparado ao experimento de validação cruzada. Os resultados em todos as bases de dados testadas foram superiores a 50%, ou seja, melhor que um classificador randômico. Entretanto, os resultados ainda não são tão bons quanto os obtidos durante os experimentos com cada base de dados individualmente e inspiram um estudo mais aprofundado.

TRABALHOS FUTUROS

As abordagens para lidar com o problema de mudança de domínio são geralmente consideradas como técnicas de *Domain Adaptation* (DA). Pesquisas recentes sobre DA estabeleceram que, quando a dissimilaridade entre domínios não é extrema, seria possível alinhar representações internas (*features*) ou adaptar as imagens de entrada de um classificador, de modo que as amostras de um domínio alvo adaptadas a um domínio de origem possam ser devidamente classificado por um classificador treinado exclusivamente com amostras do domínio de origem.

Outras técnicas (conceitualmente mais simples) podem ser usadas para aliviar essa falta de generalidade, tais como técnicas de aprendizagem de transferência supervisionada, isto é, retreinar um classificador com amostras do domínio alvo (ajuste fino). No entanto, o caso mais difícil, ou seja, a adaptação de domínio não supervisionado no contexto da análise de imagens de microscopia ótica, possivelmente tem um maior potencial de trazer avanços tecnológicos neste campo, com vasta aplicabilidade operacional.

REFERÊNCIAS

AZIMI, S. et al. Advanced steel microstructural classification by deep learning methods. *Scientific Reports*, v. 8, 2018.

BENGIO, Y. Deep learning of representations for unsupervised and transfer learning. In: *Proceedings of ICML workshop on unsupervised and transfer learning*. [S.l.: s.n.], 2012. p. 17–36.

BEZERRA, E. T. V.; AUGUSTO, K. S.; PACIORNIK, S. Discrimination of pores and cracks in iron ore pellets using deep learning neural networks. *REM-International Engineering Journal*, SciELO Brasil, v. 73, n. 2, p. 197–203, 2020.

CHEN, L.-C. et al. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014.

CHEN, L.-C. et al. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 40, n. 4, p. 834–848, 2017.

CHEN, L.-C. et al. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

CHEN, L.-C. et al. Encoder-decoder with atrous separable convolution for semantic image segmentation. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. [S.l.: s.n.], 2018.

CHENG, G. et al. When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative cnns. *IEEE transactions on geoscience and remote sensing*, IEEE, v. 56, n. 5, p. 2811–2821, 2018.

CHOLLET, F. Xception: Deep learning with depthwise separable convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2017. p. 1251–1258.

DAI, J. et al. Deformable convolutional networks. In: *Proceedings of the IEEE international conference on computer vision*. [S.l.: s.n.], 2017. p. 764–773.

DECOST, B. L. et al. High throughput quantitative metallography for complex microstructures using deep learning: A case study in ultrahigh carbon steel. *Microscopy and Microanalysis*, Cambridge University Press, v. 25, n. 1, p. 21–29, 2019.

DUAN, J. et al. Detection and segmentation of iron ore green pellets in images using lightweight u-net deep learning network. *Neural Computing and Applications*, Springer, p. 1–16, 2019.

EVSEVLEEV, S.; PACIORNIK, S.; BRUNO, G. Advanced deep learning-based 3d microstructural characterization of multiphase metal matrix composites. *Advanced Engineering Materials*, Wiley Online Library, v. 22, n. 4, p. 1901197, 2020.

GARCIA-GARCIA, A. et al. A survey on deep learning techniques for image and video semantic segmentation. *Applied Soft Computing*, Elsevier, v. 70, p. 41–65, 2018.

GLOROT, X.; BENGIO, Y. Understanding the difficulty of training deep feedforward neural networks. In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. [S.l.: s.n.], 2010. p. 249–256.

GOMES, O. Processamento de imagem digital com fiji/imagej. 08 2018.

GOMES, O.; PACIORNIK, S. Co-site microscopy–combining reflected light microscopy and scanning electron microscopy to perform ore mineralogy. In: *Ninth International Congress for Applied Mineralogy*. [S.l.: s.n.], 2008.

GOMES, O.; PACIORNIK, S. Iron ore quantitative characterisation through reflected light-scanning electron co-site microscopy. In: *INTERNATIONAL CONGRESS ON APPLIED MINERALOGY*. [S.l.: s.n.], 2008. v. 9, p. 699–702.

GOMES, O. d. F. M.; PACIORNIK, S. Multimodal microscopy for ore characterization. In: *Scanning Electron Microscopy*. [S.l.]: IntechOpen, 2012.

GOMES, O. d. F. M.; VASQUES, F. d. S. G.; NEUMANN, R. Cathodoluminescence and reflected light correlative microscopy for iron ore characterization. 2018.

HE, K. et al. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. Disponível em: <http://arxiv.org/abs/1512.03385>.

HEATON, J. *Applications of Deep Neural Networks*. 2020.

HOWARD, A. G. et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

IGLESIAS, J. C. Á.; SANTOS, R. B. M.; PACIORNIK, S. Deep learning discrimination of quartz and resin in optical microscopy images of minerals. *Minerals Engineering*, Elsevier, v. 138, p. 79–85, 2019.

IOFFE, S.; SZEGEDY, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015. Disponível em: <http://arxiv.org/abs/1502.03167>.

JIANG, F. et al. Feature extraction and grain segmentation of sandstone images based on convolutional neural networks. In: IEEE. *2018 24th International Conference on Pattern Recognition (ICPR)*. [S.l.], 2018. p. 2636–2641.

KARIMPOULI, S.; TAHMASEBI, P. Segmentation of digital rock images using deep convolutional autoencoder networks. *Computers & geosciences*, Elsevier, v. 126, p. 142–150, 2019.

KINGMA, D. P.; BA, J. *Adam: A Method for Stochastic Optimization*. 2014.

KONDO, R. et al. Microstructure recognition using convolutional neural networks for prediction of ionic conductivity in ceramics. *Acta Materialia*, v. 141, p. 29–38, 2017. ISSN 1359-6454. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1359645417307383>.

LI, J. et al. Deep fusion feature extraction and classification of pellet phase. *IEEE Access*, IEEE, v. 8, p. 75428–75436, 2020.

LING, J. et al. Building data-driven models with microstructural images: Generalization and interpretability. *Materials Discovery*, Elsevier, v. 10, p. 19–28, 2017.

LITJENS, G. et al. A survey on deep learning in medical image analysis. *Medical Image Analysis*, v. 42, p. 60 – 88, 2017. ISSN 1361-8415. Disponível em: <http: //www.sciencedirect.com/science/article/pii/S1361841517301135>.

LIU, C. et al. An enhanced rock mineral recognition method integrating a deep learning model and clustering algorithm. *Minerals*, Multidisciplinary Digital Publishing Institute, v. 9, n. 9, p. 516, 2019.

LIU, W.; RABINOVICH, A.; BERG, A. C. Parsenet: Looking wider to see better. *arXiv preprint arXiv:1506.04579*, 2015.

LIU, X. et al. Ore image segmentation method using u-net and res_unet convolutional networks. *RSC Advances*, Royal Society of Chemistry, v. 10, n. 16, p. 9396–9406, 2020.

LONG, J.; SHELHAMER, E.; DARRELL, T. Fully convolutional networks for semantic segmentation. *CoRR*, abs/1411.4038, 2014. Disponível em: <http://arxiv.org/abs/1411.4038>.

LORENZONI, R. et al. Semantic segmentation of the micro-structure of strain-hardening cement-based composites (shcc) by applying deep learning on micro-computed tomography scans. *Cement and Concrete Composites*, Elsevier, v. 108, p. 103551, 2020.

MALLAT, S. *A wavelet tour of signal processing*. [S.l.]: Elsevier, 1999.

MASCI, J. et al. Steel defect classification with max-pooling convolutional neural networks. In: IEEE. *The 2012 International Joint Conference on Neural Networks (IJCNN)*. [S.l.], 2012. p. 1–6.

MONTI, F. et al. Fake news detection on social media using geometric deep learning. *arXiv preprint arXiv:1902.06673*, 2019.

RONNEBERGER, O.; FISCHER, P.; BROX, T. U-net: Convolutional networks for biomedical image segmentation. In: SPRINGER. *International Conference on Medical image computing and computer-assisted intervention*. [S.l.], 2015. p. 234–241.

RP, K.; CL, S. An effective sem-based image analysis system for quantitative mineralogy. *KONA Powder and Particle Journal*, Hosokawa Powder Technology Foundation, v. 11, p. 165–177, 1993.

SCHINDELIN, J. et al. Fiji: an open-source platform for biological-image analysis. *Nature methods*, Nature Publishing Group, v. 9, n. 7, p. 676–682, 2012.

SHEN, D.; WU, G.; SUK, H.-I. Deep learning in medical image analysis. *Annual review of biomedical engineering*, Annual Reviews, v. 19, p. 221–248, 2017.

SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. *arXiv 1409.1556*, 09 2014.

STALLKAMP, J. et al. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural networks*, Elsevier, v. 32, p. 323–332, 2012.

SUTTON, C.; MCCALLUM, A. An introduction to conditional random fields for relational learning. *Introduction to statistical relational learning*, MIT press Cambridge, Mass., v. 2, p. 93–128, 2006.

SVENSSON, T. *Semantic Segmentation of Iron Ore Pellets with Neural Networks*. 2019.

YI, L.; LI, G.; JIANG, M. An end-to-end steel strip surface defects recognition system based on convolutional neural networks. *steel research international*, Wiley Online Library, v. 88, n. 2, p. 1600068, 2017.

ZHANG, B. et al. Convolutional neural network-based inspection of metal additive manufacturing parts. *Rapid Prototyping Journal*, Emerald Publishing Limited, 2019.

ZHANG, Y. et al. Intelligent identification for rock-mineral microscopic images using ensemble machine learning algorithms. *Sensors*, Multidisciplinary Digital Publishing Institute, v. 19, n. 18, p. 3914, 2019.

ZHAO, H. et al. Pyramid scene parsing network. *CoRR*, abs/1612.01105, 2016. Disponível em: <http://arxiv.org/abs/1612.01105>.